

TENNESSEE VALLEY AUTHORITY

CHATTANOOGA, TENNESSEE 37401
5N 157B Lookout Place

JUL 20 1988

U.S. Nuclear Regulatory Commission
ATTN: Document Control Desk
Washington, D.C. 20555

Gentlemen:

In the Matter of the Application of)
Tennessee Valley Authority)

Docket Nos. 50-390
50-391

WATTS BAR NUCLEAR PLANT (WBN) - RESISTANCE TEMPERATURE DETECTOR (RTD) BYPASS LOOP ELIMINATION AND EAGLE 21 ELECTRONICS UPGRADE - STATUS UPDATE

The purpose of this letter is to provide a status update of the RTD Bypass Loop Elimination and Eagle 21 electronics upgrade issue at WBN and to provide those items for which WBN has a complete response available.

Enclosure 1 provides response to NRC letters dated February 2 and March 25, 1987, which requested additional information with respect to proposed revisions to the Final Safety Analysis Report (FSAR), chapters 5, 7, and 15. Enclosure 2 provides response to items from the January 15 and 16, 1987 NRC audit and the revised Verification and Validation (V&V) Plan.

With regard to requested NRC action, a second audit of the RTD Bypass Elimination Eagle 21 equipment software program is presently targeted for the fourth quarter of 1988. Further details will be informally coordinated among Westinghouse, TVA, and NRC representatives. TVA expects the successful completion of the audit will result in the issuance of a Safety Evaluation Report (SER) on this issue. Summary statements of commitments contained in this submittal are provided in enclosure 3.

If there are any questions, please telephone J. A. Domer at (615) 365-8650.

Very truly yours,

TENNESSEE VALLEY AUTHORITY

R. Gridley
R. Gridley, Director
Nuclear Licensing and
Regulatory Affairs

Enclosures
cc: See page 2

Rec'd 9/28/90
FOL Per P. Tom
11
M/A 2
[Signature]

9010040207 880720
PDR ADDCK 05000390
A. PDC

U.S. Nuclear Regulatory Commission

JUL 20 1988

cc (Enclosures):

✓ Ms. S. C. Black, Assistant Director
for Projects
TVA Projects Division
U.S. Nuclear Regulatory Commission
One White Flint, North
11555 Rockville Pike
Rockville, Maryland 20852

Mr. F. R. McCoy, Assistant Director
for Inspection Programs
TVA Projects Division
U.S. Nuclear Regulatory Commission
Region II
101 Marietta Street, NW, Suite 2900
Atlanta, Georgia 30323

U.S. Nuclear Regulatory Commission
Watts Bar Resident Inspector
P.O. Box 700
Spring City, Tennessee 37381

50-390

WATTS BAR 1,2

TVA

RESPONSE TO NRC'S REQUEST
FOR ADD'L INFO

w/ltr dtd 7/20/88

#901004020

-NOTICE-

THE ATTACHED FILES ARE OFFICIAL RECORDS OF THE INFORMATION & REPORTS MANAGEMENT BRANCH. THEY HAVE BEEN CHARGED TO YOU FOR A LIMITED TIME PERIOD AND MUST BE RETURNED TO THE RECORDS & ARCHIVES SERVICES SECTION P1-22 WHITE FLINT. PLEASE DO NOT SEND DOCUMENTS CHARGED OUT THROUGH THE MAIL. REMOVAL OF ANY PAGE(S) FROM DOCUMENT FOR REPRODUCTION MUST BE REFERRED TO FILE PERSONNEL.

-NOTICE-

ENCLOSURE 1

WATTS BAR NUCLEAR PLANT (WBÑ) - UNITS 1 AND 2
RTD BYPASS ELIMINATION - RESPONSE TO NUCLEAR REGULATORY
COMMISSION'S (NRC) REQUEST FOR ADDITIONAL INFORMATION

Attachment 1 - Response to NRC's request for additional
information, as requested by
B. J. Youngblood's letter to
S. A. White dated February 2, 1987.

Attachment 2 - Response to NRC's request for additional
information, as requested by
John A. Zwolinski's letter (NRC) to
S. A. White dated March 25, 1987.

Response to NRC Request for Additional Information
RTD Bypass Modification
Watts Bar Nuclear Plant

March, 1987

July, 1987 Revision 1

November, 1987 Revision 2

References:

1. Summary of Meeting to Discuss RTD Bypass System Removal at Watts Bar, dated October 23, 1986.
2. Letter from J. Domer, TVA, to B. J. Youngblood, NRC, dated December 1, 1986.
3. Letter from B. J. Youngblood, NRC, to S. A. White, TVA, dated January 16, 1987.
4. Letter from B. J. Youngblood, NRC, to S. A. White, TVA, dated February 2, 1987.

Summary

The following questions are from Reference 4.

- Q.1 In the December 1, 1986, submittal, it was stated that the three hot leg RTDs are electronically averaged to provide the T_{hot} signal for use by protection and control systems. It is the staff's understanding if one of the RTDs fails, you can automatically add a bias to the other two readings to simulate the variation that exists because of streaming. Please describe this process more fully. Also, provide information regarding when the failed RTD will be replaced. Will information on this replacement be included in the Tech Spec or is it already incorporated within an existing Tech Spec?

WESTINGHOUSE CLASS 3

A.1 The input bias that is used to compensate "T-hot average" upon loss of one narrow range T-hot signal is based upon "T-hot average" with three valid RTD inputs. There is one bias value associated with each narrow range T-hot RTD input signal. Simply stated, the bias value for each RTD is calculated while all three RTD's are considered to be valid by subtracting the average of the remaining two RTD's from the "T-hot average" value for that loop. Then, if a RTD should fail, "T-hot average" for that loop is calculated by adding the bias value for the failed RTD to the average of the remaining two RTD's. This formula ensures that the calculated value of "T-hot average" with two valid RTD's is nearly identical to the value of "T-hot average" that was calculated with three valid RTD's.

If a single RTD does fail, the value of "T-hot average" would be calculated as described above and a status light indicating "trouble" would be activated in the control room. The failed RTD would be replaced during a subsequent plant outage. It should be noted that no information on the replacement of a single failed RTD needs to be incorporated into the Watts Bar Technical Specifications since the plant's setpoint methodology and safety analyses only assume two operational T-hot RTD's in each loop.

If two or three hot leg RTD's in the same loop fail, a dedicated alarm and annunciator would be activated indicating a failed channel. Technical Specification Table 3.3-1 details the action which must be taken for a failed ΔT /OP ΔT channel.

Q.2 The staff requires more information relative to the structural integrity of the narrow range and wide range Reactor Coolant System (RCS) temperature sensors (RTDs) which will be mounted in thermowells that protrude into the RCS hot and cold legs (Reference letter from J. A. Domer to Director of NRR, "Watts Bar Nuclear Plant - RTD Bypass Loop Elimination/Utilization of Eagle 21 Electronics - FSAR Chapters 5 and 7," dated December 1, 1986). Provide a description and summary of results of

analyses and/or tests which were performed to demonstrate that the proposed thermowells will withstand all anticipated flow induced vibration loads in combination with all other loads which are identified in applicable portions of FSAR Section 3.9.3. Include a discussion of the possibility of thermowell wear and high cycle fatigue damage which could be caused by turbulent buffeting and/or vortex shedding. Describe what post-installation tests, if any, would be performed to ensure that acceptable margins exist to prevent local fluid flow velocities from producing turbulent flow loads in resonance with the natural frequencies of the temperature probes.

- A.2 See attached report, entitled, WAT/WBT Thermowell Structural Evaluation, dated July, 1987, Revision 1. This report describes and summarizes the analyses performed to demonstrate the structural adequacy of the thermowells when subjected to pressure loadings, hydraulic steady flow loads, vibratory loads due to turbulent buffeting, vortex shedding, pump pulsations and seismic excitations, and post-installation tests are not considered to be necessary.

WAT/WBT THERMOWELL STRUCTURAL EVALUATION

FEBRUARY, 1987

JULY, 1987, REVISION 1

SUMMARY

This report provides descriptions and summary of the analyses performed to demonstrate the structural adequacy of the thermowells when subjected to flow induced vibration loads. Loads other than flow induced vibration loads such as pressure, hydraulic steady flow and seismic loads are also included in the evaluation of thermowell structural adequacy.

Displacements and stresses due to each of the loads are calculated to determine if wear or fatigue causes any problem to the thermowells.

An analysis of the vibratory loads induced by turbulence buffeting demonstrates that these loads are too small to cause high cycle fatigue of the thermowells. Similarly, the displacement amplitudes are calculated to be on the order of 0.0005 inch and as such, the likelihood of thermowell wear due to contact with other nearby surfaces is extremely small.

Turbulent buffeting was considered to be the dominant cross-flow induced vibration mechanism because the flow Reynold's numbers are between 1.24×10^6 and 1.44×10^6 (i.e. in the aperiodic regime of cross-flow excitation). Nevertheless, fatigue loads and displacement amplitudes were calculated, assuming vortex-shedding excitation. The results here also indicated that high cycle fatigue and wear were insignificant.

An analysis for pump pulsation loads was also carried out to determine its effects on the thermowells. The results indicated that the loads are quite small and as a result the stress and displacement responses of the thermowells were insignificant. Hence, for reasons outlined above, fatigue and wear will be of no concern.

Based on the analysis performed, it is concluded that neither high cycle fatigue nor wear for any plausible forced induced vibration mechanism is a problem for the thermowells and its structural integrity will be maintained.

INTRODUCTION

This report covers the load development and structural analyses of the RTD Bypass Elimination Thermowell. As the thermowells are pressure retaining components in the primary coolant loop, it was analyzed to the requirements of the ASME Boiler and Pressure Vessel Code, Section III, Subsection NB. The structural integrity of the thermowell has been examined for all normal operating conditions. The reaction forces and moments at the thermowell welds are also calculated.

DESIGN LOADING CONDITIONS

The thermowell was considered for the following design loads for all normal operating conditions;

- A) Pressure Loadings
- B) Hydraulic Steady Flow Loads
- C) Vibratory Loads due to Turbulent Buffeting
- D) Vortex Shedding
- E) Pump Pulsations
- F) Seismic Excitations

CRITERIA

The requirements of the ASME B&PV Code Section III, Subsection NB are used as the criteria to evaluate the stress levels in the thermowells for all loading conditions considered. The stress allowables are given in the following table.

Allowable Stress Intensity

	Pm	Pm+Pb
Level A & B	1.0 Sm	1.5 Sm

Where Sm = 16,700 psi for the SA-182, F316 Stainless Steel at 650 °F.

The concern of metal fatigue was also evaluated using the ASME fatigue curves of I-9.2 and I-9.2.2 of Appendix I for the low and high cycle fatigue, respectively.

DRAWING LIST

The thermowells are installed on both the hot and the cold leg pipes connecting to the reactor outlet and inlet nozzles, respectively. The design drawings used for this analysis are listed as follows:

Table 1 Thermowells and Corresponding Installation Drawings

Location	Thermowell	Installation
a) Hot Leg	1847E84H02 Rev. 3	9558D95 Rev. 2
b) 2" Cold Leg	1847E83H02 Rev. 3	1871E46 Rev. 3
c) Cold Leg	2326D52H01 Rev. 2	1863E26 Rev. 4

METHODOLOGY

Since the thermowells installed on the cold legs, item c), are diametrically identical except slightly longer than those installed on the hot legs, item a), they are selected, for the convenience of structural analysis, to represent the "hot leg" thermowells. In item b), since the WBT thermowells extend more into the flow field than the WAT thermowells, they are used to represent all "cold leg" thermowells. These two "representative" thermowells are shown in the following figure.

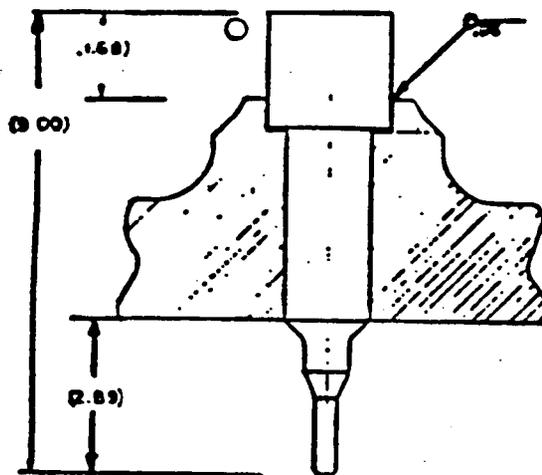
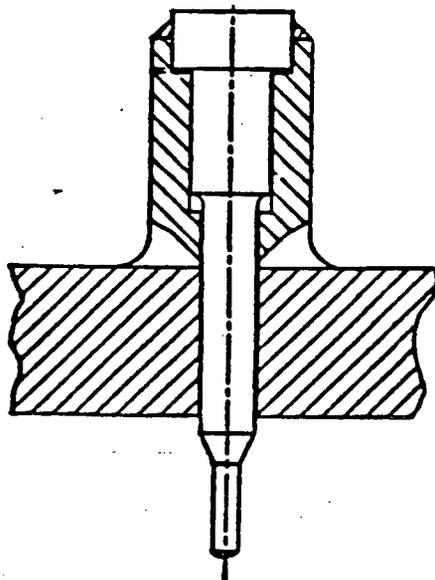


Fig. 1 COLD Leg Thermowells

Cold Leg Thermowells

From the thermowell geometries shown above, it is obvious that the most critical sections are the root of the 0.4" tip sections where the thickness are the smallest. Therefore, it is suffice to analyze the stresses at this section only.

A) Pressure Loadings

Hoop and Radial stresses in the 0.4" tip section for the pressure loadings, 2250 psi, are calculated using the thick-wall cylinder equations. The pressure induced axial stresses are also calculated. The stresses induced by the pressure loading are the only primary membrane stresses in the section.

B) Hydraulic Steady Flow Loads

The thermowells are subjected to steady drag loads during reactor operation. The load on the 0.4" section is calculated using the following formula:

$$F_d = C_d * (\rho V^2 / 2g) * A$$

Where C_d = Drag Coefficient
 = 1 (Assumed for Conservatism)
 ρ = Water Density at Temperature

V = Flow Velocity
 A = Projected Area of Cylinder
 g = Acceleration due to gravity

For conservatism, flow velocity at pump overspeed, which is an upset condition, was used.

C) Vibratory Loads due to Turbulent Buffeting

As the thermowell tip is situated in a turbulent flow field, there are random vibratory flow forces acting upon it. The thermowell's natural frequencies were calculated with finite element models. The lowest fundamental frequency has been determined to 832 HZ. The random lift forces due to turbulent flow field that act on the tips of the thermowells are calculated using the following formula (Based on Y.C.Fung's paper "Fluctuating Lift and Drag Acting on a Cylinder in a Flow at Supercritical

Reynold's Numbers" in Journal of the Aerospace Sciences, Volume 27, Number 11, November 1960, pages 801-814)

$$F_L = \left(\frac{\pi Q}{2}\right)^{\frac{1}{2}} C_L \left(\frac{D}{L}\right)^{\frac{1}{2}} \left(\frac{\rho D}{2g}\right) [S F(S)]^{\frac{1}{2}} V^2$$

Based on the same paper, the mean displacement due to turbulent buffeting is :

$$Y = \left(\frac{1}{8} \left(\frac{\rho V^2 A}{2g}\right)^2 C_L^2 \left(\frac{D}{V}\right) [F(S)] \frac{1}{QM^2 (2\pi f)^3}\right)^{\frac{1}{2}}$$

Where C_L = Lift Coefficient
 = 1 (Assumed for Conservatism)
 L = Length of the Thermowell Tip
 = 1.5"
 f = Fundamental Frequency = 832 HZ
 Q = Amplification Factor
 = 50 (1% damping)
 S = Dimensional Frequency = $f * D / V$
 $F(S)$ = Empirical Function of S
 = $4.8 \{ [1 + 3(4.8 * \pi * S)^2] / [1 + (4.8 * \pi * S)^2]^2 \}$
 M = Cylinder Mass
 D = Cylinder Diameter in inch
 V = Fluid Velocity
 = 56.7 fps
 A = Projected Area of Cylinder
 ρ = Density of Water at the Required Temperature
 g = Acceleration due to Gravity

It is found from the calculations that mean displacement is of the order of .0005 inch. As the amplitude of displacement is very small, the likelihood of thermowell wear due to contact with other nearby surfaces is not possible. Thus, we conclude that turbulent buffeting would not cause any wear.

D) Vortex Shedding

Vortex shedding on the tip section was ruled out as a concern because the Reynold's Number was calculated to be in between 1.24×10^6 and 1.44×10^6 . Thus, no periodical shedding can occur as evidenced from the attached curve (Attachment 1).

Even if we assume there is some vortex shedding, the maximum vortex shedding frequency possible ($f = S * V / D$)

where S = Strouhal Number, conservatively taken as 0.2, V = Stream Velocity, and D = Cylinder Diameter) will be 341 Hz. This is much less than the lowest natural frequency of 832 Hz. Thus any vortex shedding load would act almost like a static load. Since vortex shedding lift co-efficients are about half as big as steady-state drag co-efficients, the vortex shedding load amplitudes are about half as big as steady-state drag loads (maximum steady-state drag load is 12.3 lbs). As the lift force of 20 lbs due to turbulent buffeting produces displacement of the order of .0005 inch, displacement due to this vortex shedding load (maximum amplitude of about 6.20 lbs) will be even less and thus wear due to this event will be of no concern.

E) Pump Pulsations

The fundamental frequencies of the thermowells are far greater than the pump pulsating frequencies (19.7, 173 and 277 Hz). Higher modes which may be closer to fundamental frequencies of thermowell will not contribute significantly to its responses because of their lower participation factors.

Even if we assume that pump induced vibrations exert a load on the thermowell, its magnitude will be on the order of:

$$\text{load} = \Delta p * A * 2 f * D/C$$

Where Δp = Pressure fluctuation at the pump (maximum of 1.1 psi regardless of the pump frequency as per "An Experimental Investigation of Reactor Coolant Pump Induced Pressure Fluctuations" - L.A.Shockling & P.J.Sowatsky, WCAP - 10476, December 1983, Westinghouse Proprietary Report.)

A = Projected area (assume 3 in², although only 0.6 in², thus using a factor of 5)

f = Pump frequency
= 277 Hz (maximum)

D = Diameter of the cylinder
= 0.87 inch

C = Speed of sound in Fluid

The magnitude of pump induced vibration loads in this case will be of the order of .0005 lb which can be neglected. Thus the displacement due to this load, even

with amplification of 20, will be negligible and thermowell will have no problem due to wear. Similarly, the stresses due to this load is also very small and hence fatigue usage factor is negligible. Hence, there is no concern for any adverse effects due to pump induced vibration.

F) Seismic Excitations

As the thermowells are rigid components, seismic excitation force can be calculated by multiplying the zero period acceleration (ZPA) by the corresponding masses and statically applied to the tip section. The magnitudes of ZPA have been conservatively assumed at 2g in both the horizontal and the vertical directions. The resulting forces have been found to be negligible.

The forces calculated from items B) through F), were summed up and applied to the tip section, the corresponding bending stresses at the root of the 0.4" tip section were calculated.

RESULTS OF STRESS ANALYSIS

The results of stress analysis, in terms of margins of safety, are tabulated as follows:

Category	Allowable	Cold Leg	Hot Leg
Pm	1.0 Sm	3.4	3.4
Pm + Pb	1.5 Sm	2.6	2.0

Using the results of the stress analysis, usage factors are calculated to check the effects of fatigue on the thermowells and is found that it is insignificant.

CONCLUSIONS

To ensure structural adequacy of the thermowells, we have analyzed them for the loads due to a) Pressure, b) Hydraulic Steady Flow, c) Turbulent Buffeting, d) Vortex Shedding, e) Pump Pulsations and f) Seismic Excitations. Based on the analyses reported in the previous sections, we found that loads resulting from them are small. Hence, the amplitude of displacements are even smaller and would not move the thermowells far enough to wear against other surfaces. Similarly, the stresses due to those loads are also

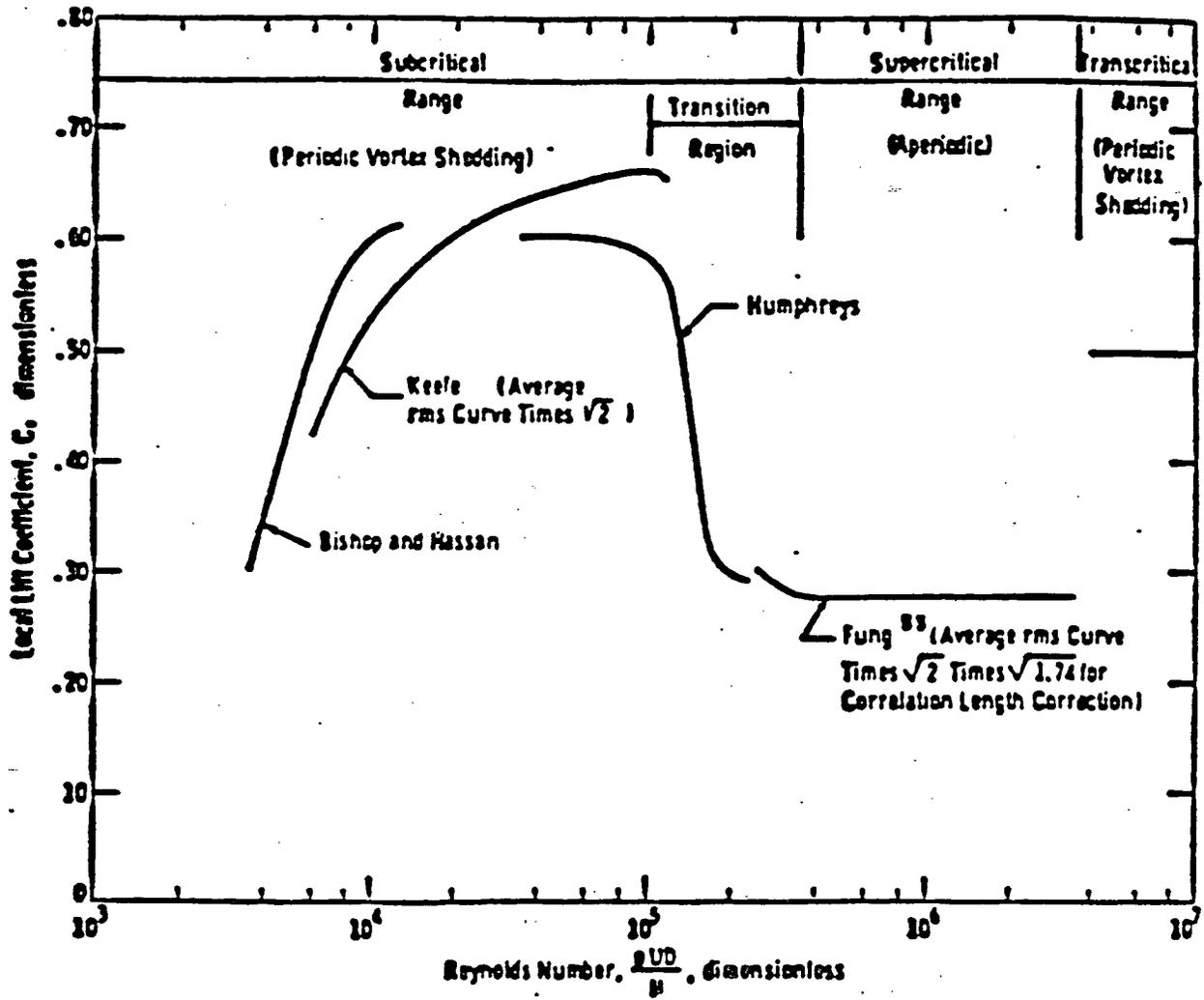
WESTINGHOUSE PROPRIETARY CLASS 3

small and would not cause any fatigue problem for the thermowells.

Thus, evaluation of thermowells, when subjected to above loads, shows that the loads and displacements resulting from them are small enough and neither high cycle fatigue nor wear is of any concern.

WESTINGHOUSE PROPRIETARY CLASS 3

ATTACHMENT 1



LIFT COEFFICIENT vs. REYNOLDS NUMBER

ATTACHMENT 2

Response to NRC Request for Additional Information

RTD Bypass Modification

Watts Bar Nuclear Plant

April, 1987

November, 1987 - Revision 1

References:

1. Summary of Meeting to Discuss RTD Bypass System Removal at Watts Bar, dated October 23, 1986.
2. Letter from J. Domer, TVA, to B. J. Youngblood, NRC, dated December 1, 1986.
3. Letter from B. J. Youngblood, NRC, to S. A. White, TVA, dated January 16, 1987.
4. Letter from B. J. Youngblood, NRC, to S. A. White, TVA, dated February 2, 1987.
5. Letter from John A. Zwolinski to S. A. White, dated March 25, 1987.

Summary

The following represents the NRC statements and questions from Reference 5, above.

Questions on Watts Bar RTD Bypass Loop Removal

References

1. Meeting Summary from T. J. Kenyon, NRC to NRC Staff Attendees, dated October 23, 1986.
2. Letter from R. Gridley, Tennessee Valley Authority, T. B. J. Youngblood, NRC, dated January 27, 1987.

WESTINGHOUSE CLASS 3

The Reactor Systems Branch has reviewed the above references from a thermal hydraulic viewpoint in regards to the RTD bypass loop removal. Reference 1 is a summary of the meeting with the Tennessee Valley Authority (TVA) on October 14, 1986, with representatives of NRC, TVA, and Westinghouse to discuss TVA's proposal to remove the RTD bypass system at Watts Bar and includes a copy of TVA's presentation as an enclosure. Reference 2 provides marked up pages for accident analyses in Chapter 15 affected by the RTD bypass removal. These include uncontrolled bank withdrawal at power, loss of load/turbine trip, and RCS depressurization.

Q.1 It is noted that the modified scoop (Reference 1) for the RTD thermowell is cut back so that the RTD is directly exposed to the flow rather than receiving flow through holes in the scoop. Is the temperature sensed at a radial dimension equivalent to the middle hole of the original scoop or at a distance which would give the true weighted average value. (It is noted that holes at a greater radius represent a larger flow area). Is there a turbulence effect from the edge of the cutoff scoop that would affect the accuracy of the RTD sensor value?

R.1 The heat transfer sensitive tip of the thermowell is located at a radial dimension that is essentially the same as the radial location of the center hole of the scoop before it was removed. This location was selected so that the temperature measured by the thermowell RTD would be the same as the average temperature of the sample flow that would have been collected by the scoop. The location of the thermowell tip or the middle scoop hole is on or close to the radius of a circle that divides the area of the pipe into two equal parts. Locating the temperature measurement point at this radius has been shown by analysis to provide the most accurate average temperature measurement for any hot leg temperature streaming distribution. On the Watts Bar application, the tip of the thermowell is located approximately 1.5 inches from the end of the remaining part of the scoop. The fluid velocity will increase slightly past the thermowell as well as the scoop stub, but this velocity increase should not have any effect on the accuracy of the temperature measurement.

WESTINGHOUSE CLASS 3

- Q.2 Table 15.1-3 of Reference 2 shows that the time delay assumed in the accident analysis for the trip function for overtemperature delta T and overpower delta T is 7.0 seconds. Reference 1 states that although the time delay is 6.5 seconds, 7.0 seconds is used for conservatism. In a similar RTD bypass loop modification for another plant, it was reported that the measured response time was found to be as high as 11.5 seconds instead of 6.5 seconds. Is Watts Bar able to confirm the RTD response time value of 6.5 seconds?
- R.2 Laboratory testing is underway aimed at identifying the root causes of the higher response times measured. To date, three variables have been identified: 1.) the radial gap between the RTD sensor and the thermowell, 2.) the fit between the RTD sensor tip and the bottom of the thermowell, and, 3.) the thermowell straightness. Two possible solutions are under consideration: 1). silver plating of the RTD sensor tip, and, 2.) applying a heat transfer metallic grease to the RTD sensor tip. Laboratory testing of silver plating of the RTD sensor tip has been conducted and the results support the response times expected for Watts Bar.

Additionally, field testing of silver plated RTD sensor tips has been performed. The silver plating minimizes the radial air gap between the RTD sensor tip and the thermowell. Houston Light & Power performed an in-situ response time test of their thermowell mounted RdF RTDs at the 250°F heatup plateau during startup at South Texas 1. The test was performed by the Analysis & Measurement Services (AMS) Corporation using the Loop Current Step Response (LCSR) method. Sixteen (16) RTDs were tested with only two (2) exhibiting response times greater than 5.5 seconds. The response times of these two RTDs were 5.7 and 5.9 seconds, with the rest of the RTDs falling in a range between 4.0 and 5.5 seconds. The two slower RTDs had radial gaps larger than the average of the remaining RTDs and the reduction of that gap would be expected to result in a response time less than 5.5 seconds.

Additional data will be taken at South Texas 1 when the plant reaches the hot standby plateau (567°F). This data will serve to confirm the 250 degree data as well as discover any temperature related response time improvements. In any event the data taken to date provides confidence that the silver plated RdF RTDs can provide satisfactory response times.

Q.3 FSAR pages 15.2-8 and 15.2-25 (Reference 2) have a modification insert - "pressurizer pressure - 46 psi allowance for steady state fluctuations and measurement error." Has this value been modified because of the RTD bypass removal? Is there any affect from the RTD bypass removal on the accuracy and value of the RCS average temperature? If so, what is the change and has this affected the reactor protection system setpoints?

R.3 The pressurizer pressure allowance for steady state fluctuations and measurement error which was used in the safety analyses was not modified due to the RTD bypass removal. This uncertainty allowance (46 psi) reflects the uncertainty associated with the Barton pressure transmitters used in the Watts Bar plants.

As part of the RTD Bypass Loop Removal, the Rosemount RTD's which are currently in the Watts Bar plant are being replaced with RdF RTD's. The RdF RTD's have a temperature uncertainty of 1.2°F which represents an increase of 0.5°F over the 0.7°F temperature uncertainty associated with the Rosemount RTD's. As a result of this change in RTD types, the RTD Bypass Removal does have an affect on the accuracy of the RCS average temperature measurement. However, in all of the Watts Bar non-LOCA safety analyses, an additional 2.5°F temperature uncertainty for margin has always been included. As a result, the 0.5°F increase in uncertainty was absorbed into this margin and did not affect the safety analyses or the reactor protection system setpoints.

WESTINGHOUSE CLASS 3

Q.4 For the FSAR Chapter 15 accident reanalysis, you have presented information on the following:

- a. Uncontrolled bank withdrawal at power (Figures 15.2-4 to 8)
- b. Loss of load/turbine trip (Figures 15.2-19 to 26)
- c. RCS depressurization (Figures 15.2-37 to 39)

Please provide a discussion of the results comparing the affects from before and after the RTD bypass removal and justify their acceptability. It is noted that in the DNBR vs time curve in Figure 15.2-5, the DNBR value is very close to the 1.30 limiting value. It is difficult to tell if the value is at, slightly above or below 1.30. If it is above, has the correct uncertainties for the new RTD and flow measurement analysis been included? Reference 1 indicated that the uncontrolled boron dilution accident would be reanalyzed. The results of this analysis were not in Reference 2. Please provide the results and the discussion for justifying its acceptability.

R.4 The DNBR value shown in Figure 15.2-5 remained above the 1.30 limiting value throughout the transient, and the correct uncertainties for the new RTD and the flow uncertainty were included. As discussed in the response to question 3, the additional temperature uncertainty associated with the new RTD was absorbed into the 2.5°F uncertainty margin available in each analysis. Similarly, a preliminary flow measurement analysis has shown that the current uncertainty value of 1.8% remains applicable and was included in the safety analyses.

No FSAR markups were provided for the uncontrolled boron dilution at power accident because the total time available for operator action currently reported in the Watts Bar FSAR bounds the results obtained from the reanalysis of the event.

ENCLOSURE 2

VERIFICATION AND VALIDATION PLAN FOR THE
EAGLE 21 SYSTEM UTILIZATION - RESPONSE SUMMARY

J. A. Zwolinski's (NRC) letter to S. A. White dated April 27, 1987, contains the Nuclear Regulatory Commission's (NRC) audit report on the Verification and Validation (V&V) Plan for the Eagle 21 system utilization. The audit report contained three open items as outlined below:

Item 1: Confirm that the V&V Plan has been executed as described.

Response: This will be answered by a draft V&V report, the final NRC audit, and the final V&V report reflecting the results of the audit.

Item 2: Verify that independence was present during the formal design verification phase.

Response: Provided as attachment 1.

Item 3: Either classify all software residing with the Eagle 21 mainframe as class 1E software or provide acceptable justification for this software being classified as nonclass 1E.

Response: Revised sections 5.4, 5.4.3, 5.4.3.1, and 5.4.4.1 of revision 2 of the subject V&V Plan contain concise criteria for the nonsafety-related software (see attachment 2).

ATTACHMENT 1

NRC Concern Number 2

The Eagle-21 Design, Verification, and Validation Plan does not appear to provide for acceptable independence during the software verification process.

On page 10 of the referenced Audit Report on Verification and Validation Plan for the Eagle-21 System Utilization, dated April 27, 1987, it is stated, "The staff believes that the requirements of Appendix B of 10 CFR 50 take precedence over standards cited by the manufacturer in this area. Appendix B states in part that persons and organizations performing quality assurance functions shall report to a management level such that the required authority and organizational freedom include independence from cost and schedule."

Response:

Computer Software Verification and Validation for safety-related computer systems is recognized by Westinghouse as an important activity requiring attention to the assignment of capable personnel and attention to the management of those personnel such that they are encouraged to find and report all discrepancies. However, there is no fundamental difference between the Verification and Validation of computer software and the verification/checking of engineering design for safety-related equipment which has been done by Westinghouse for three decades. It has been and remains the policy and practice of Westinghouse to insist that checkers/verifiers be 1) qualified technically to perform the work being checked, 2) different person(s) from those who performed the work and (3) organizationally free to do their checking properly. It is not required by Westinghouse policy or procedure that any particular organizational structure be imposed to achieve the required level of independence when competent professionals are involved. This Westinghouse policy practice both predates 10 CFR 50 Appendix B and has been found in hundreds of instances to satisfy the requirements of 10 CFR 50 Appendix B as well as the relevant industry standards.

10 CFR 50 Appendix B, Section I recognizes, "Because of the many variables involved, such as the number of personnel, the type of activity being performed, and the location or locations where activities are performed, the organizational structure for executing the quality assurance program may take various forms provided that the persons and organizations assigned the quality assurance functions have the required authority and organizational freedom." Section II of 10 CFR 50 Appendix B states, "This program shall be documented by written policies, procedures or instructions and shall be carried out throughout plant life in accordance with those policies, procedures, or instructions." Section III of 10 CFR 50 Appendix B states, "The verifying or checking process shall be performed by individuals or groups other than those who performed the original design, but who may be from the same organization."

Westinghouse has had in place for many years a comprehensive Quality Assurance Program which has been reviewed and approved at each revision by the NRC and its predecessor, AEC. This program clearly identifies the various organizational responsibilities for satisfying the requirements of 10 CFR 50 Appendix B. Section 17.1.3 (Design Control) of the Westinghouse Electric Corporation Water Reactor Divisions Quality Assurance Plan states, "The design verification method is selected based on the complexity of the design and on the type of design document being verified and is performed by individuals or groups other than those who performed the original design." The Eagle-21 Design, Verification, and Validation Plan clearly satisfies the requirements of both 10 CFR 50 Appendix B and the Westinghouse Quality Assurance Plan. The Westinghouse Quality Assurance Plan (WCAP-8370, Revision 10A) has most recently been reviewed by the NRC staff in August 1984 and found to be acceptable as documented in a letter from J. Nelson Grace, Division of Quality Assurance, Safeguards, and Inspection Programs, Office of Inspection and Enforcement to Mr. E. P. Rahe, Jr., Manager, Nuclear Safety, Water Reactors Divisions, Westinghouse Electric Corporation, dated August 29, 1984.

DESIGN SPECIFICATION 408A47	DATED 11/7/86	REVISION NO. 2	DATED 2/25/87	ORIGINAL ISSUE <input type="checkbox"/>	SUPERSEDES PREVIOUS REVISIONS <input checked="" type="checkbox"/>
--------------------------------	------------------	-------------------	------------------	--	--

PROJECT: Generic

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

SHOP ORDER: 322/393

SYSTEM: Process Protection System

ATTACHMENTS

Reviewed by: SE Lang / Z E. Eric 11-18-86
Nuclear Safety

REV. 1 SE Lang / Z E. Eric 01-09-87

REV. 2 SE Lang / Z E. Eric 02-26-87

- 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	11/18/86	1/13/87	2/27/87			
SHOP ORDER HOLDER	C.E. Corl J.B. Waclo	11/18/86	1/13/87	2/27/87			
MANAGER	D.P. Adomaitis	11/18/86	1/13/87	2/27/87			
PRODUCT ASSURANCE	B.F. Barnett	11/18/86	1/13/87	2/27/87			
PROJECT MANAGER	W.C. Gangloff	11/18/86	1/13/87	2/27/87			

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 ensure 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 multidisciplinary 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 be applied to all software units. If all of the following conditions are met, the software is level 2; level 1 will be used otherwise.

1. FUNCTIONS

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

2. CONNECTIONS

- a. There is no direct path to level 1 software functions via a common bus structure.
- b. There is no direct path to hardware I/O used by level 1 software functions.
- c. Data link transmission to level 1 software functions is prevented by hardware design.

3. ORGANIZATION

- 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 re-applied 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.

2

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 all 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 ROM 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. Multiple 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 requirements. 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 satisfied.

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-requirement must be uniquely identified. The test procedure generated to test each sub-requirement must be coorespondingly identified for ease of cross-referencing.

6.2.4 Abnormal-Mode Testing

During this phase of Validation the functional requirements 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 Specifications 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 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)
- 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

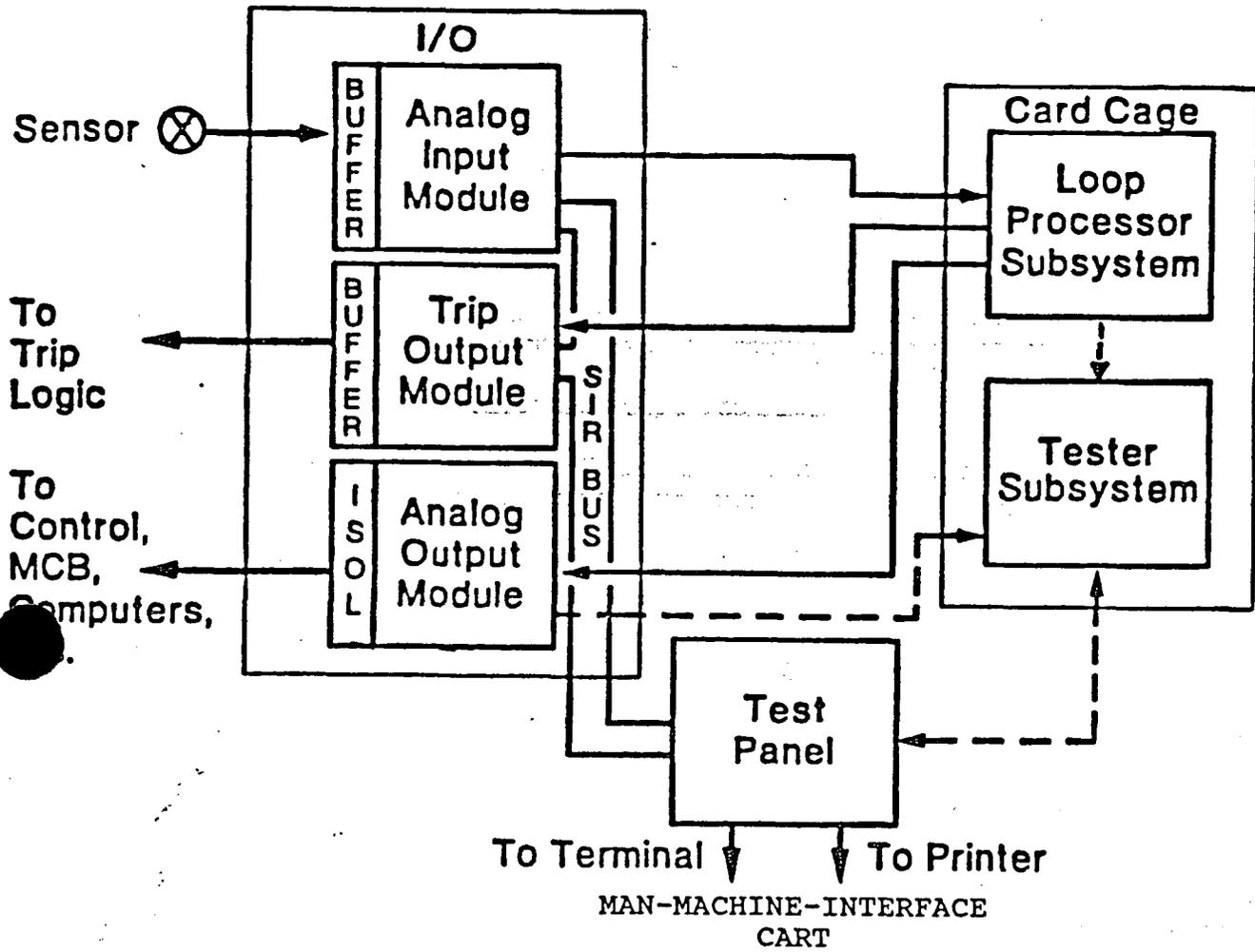
- a. Write Validation test procedures
- b. Oversee Validation testing and review test results
- c. Generate Validation Trouble Reports

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

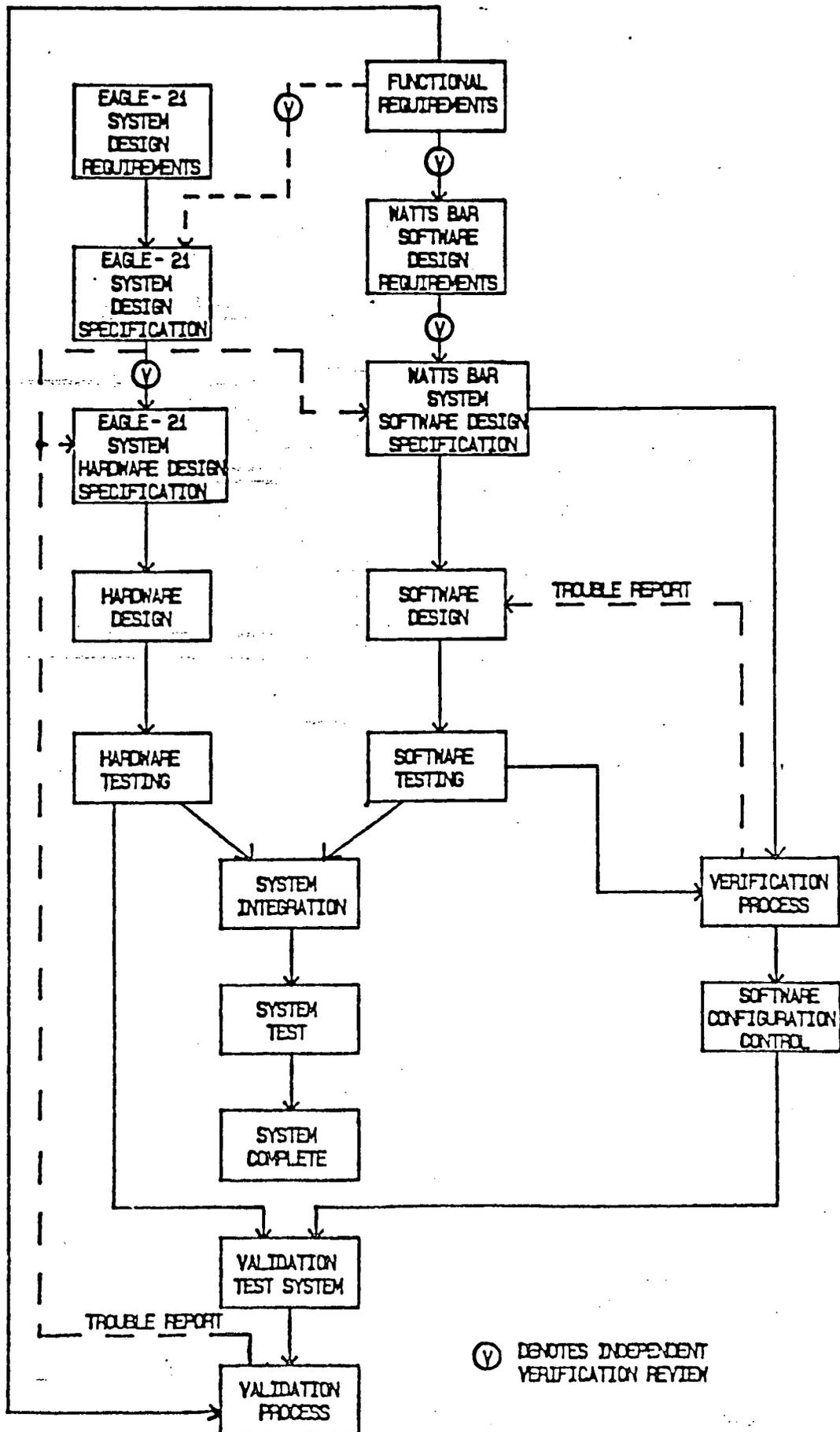
FIGURE 1

FIGURE 2
DESIGN, VERIFICATION AND VALIDATION PROCESS

DEFINITION

DESIGN

IMPLEMENTATION
AND
TEST



Ⓢ DENOTES INDEPENDENT VERIFICATION REVIEW

SOFTWARE VERIFICATION PROCESS
TABLE 1

	<u>Verification Level</u>	
	Level 1	Level 2
> FORMAL LIBRARY		
- 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 all 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.

ENCLOSURE 3

Commits to schedule a second audit for fall of 1988.