U.S. NRC RIC 2014, March 11-13, 2014

TH27: Safety Critical Software – International Perspectives

S/W Qualification Activities for Safety Critical Software

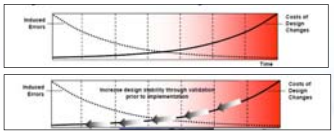
March 13, 2014

Gee-Yong Park

한국원자력연구원
Korea Atomic Energy Research Institute

Characteristics of Software Defects

- **Design Fault**
  - **Mostly introduced at the design stage**
  - **Deterministic, not random fault**

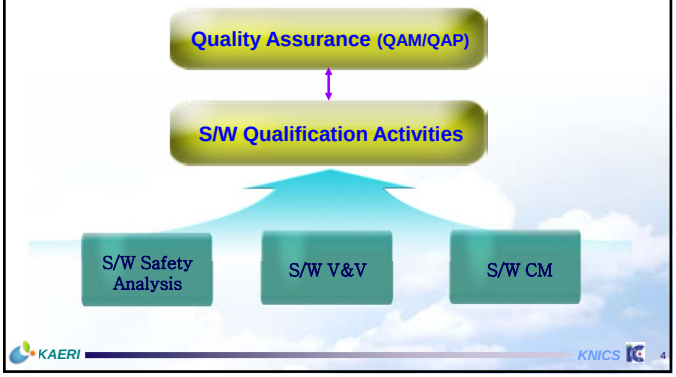


- **Aspects for Improving Software Quality**
  - **Development Process**
  - **Correct Product**
  - **Competent Organization**

Purpose of S/W Qualification Activities

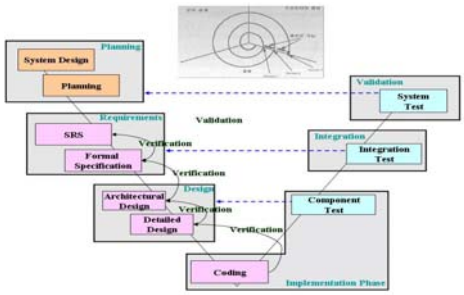
- **Target System**
  - **Application Software for Digital Reactor Protection System (Safety-Critical Class)**
  - **Project Period: 2001 – 2008 (7 year project)**
- **Purposes of S/W Qualification Activities**
  - **Satisfaction of Licensing Criteria**
    - ✓ **Evaluation of Licensing Suitability**
  - **Improvement of S/W Quality**
    - ✓ **Finding S/W Defects at an Early Stage**
    - ✓ **Finding S/W Hazards along SDLC**
    - ✓ **Adherence of S/W Development Baseline**

Main Activities of DRPS S/W Qualification

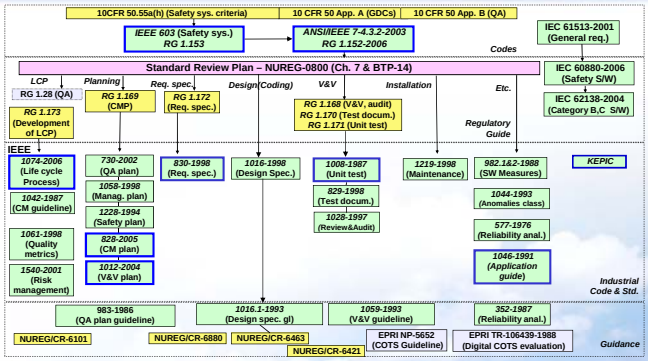


Software Lifecycle Model

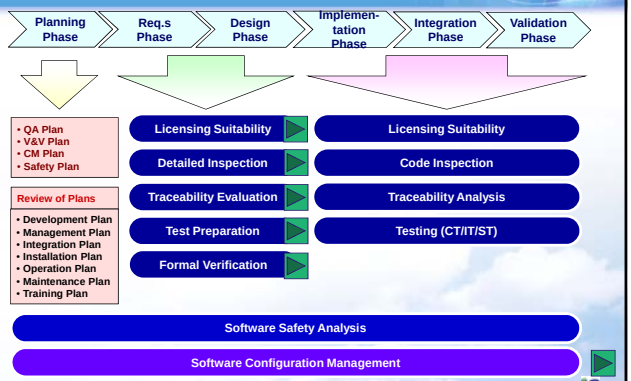
- SLCM: A Spiral Circulation of Waterfall Model
- SLC Phases: Phases in NUREG-0800 BTP/HICB-14
  - Planning, Requirements, Design, Implementation, Integration, Validation, Installation, Operation & Maintenance



Codes & Standards for DRPS S/W



S/W V&V Activities for DRPS Software



BTP: Branch Technical Position

Licensing Suitability Evaluation

- Purpose: Confirms whether or not the software requirements or design descriptions possess required software properties
- Checklist-Based Evaluation for Criteria of Software Properties (from NUREG-0800 BTP/HICB-14 & IEEE Std. 7-4.3.2)

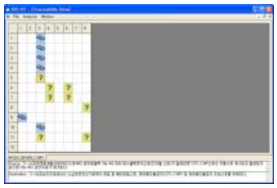
Detailed Inspection (Fagan Inspection)

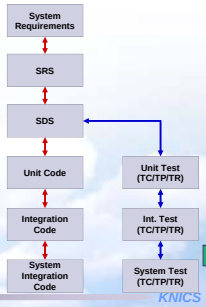
- Evaluating SRS/SDS with respect to Completeness, Consistency, Correctness
- A Checklist-Based Inspection for Functions, Inputs/Outputs, and Interfaces for Each of CCC.

Item	R.G./Code/Std	Inspection Items		Inspection Results
Completeness of S/W Function Definitions	BTP HICB-14, Chap. 3.3a	Are all operational modes S/W shall perform described ? Are all S/W functions and behaviors to be executed in each operational mode described ?	Operation Bypass	NOT complete is CPC-CWP
			Setpoint Settings	NOT complete is trip hysteresis function in the manual reset variable setpoint module
			Test Mode	NOT complete is BP1 PT scheduler

Traceability Analysis

- **Scope and Type**
  - **From system requirements to validation phase**
  - **Bidirectional (Forward/Backward) Traceability Analysis**

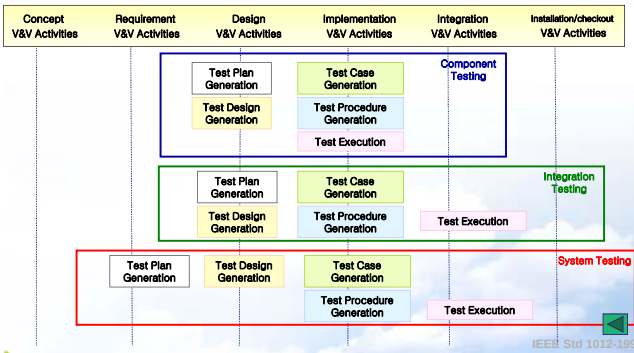




Formal Specification & Verification

- ❑ Proprietary Tools of Formal Specification & Verification
  - ➢ Requirements: NuSRS (based on NuSCR) + SMV Model Checking
  - ➢ Design: FBD Verifier
- ❑ Application Scope
  - ➢ S/W modules that take an important trip function
  - ➢ Formal verification tools usually cannot handle the entire S/W modules because of its capacity limitation of internal states
- ❑ Merits & Demerits
  - ➢ It can find a subtle S/W defect that cannot be indentified from document evaluation.
  - ➢ It is very difficult to use the tool and interpret its output.
- ❑ Automatic Code Generation
  - ➢ Necessary to reduce human coding errors → but, bulky

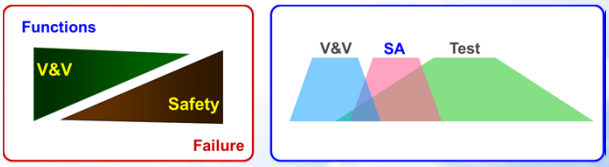
Testing Activities on SWLC Phases



IEEE Std 1012-1998

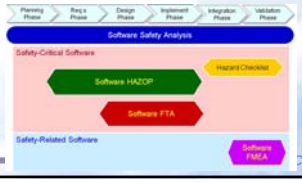
Software Safety Analysis

- Purposes of S/W V&V and S/W Safety Analysis (SSA)
  - S/W V&V: Functional/Process Requirements
  - SSA: Safety Requirements (To find out a potential hazard)
- SSA discovered defects that had not been identified in the V&V works → It can compensate for V&V works.
- It can indicate on which the testing resources will focus.



Activities of S/W Safety Analysis

- **Activities**
  - Identification of Software-Contributable System Hazards & Interface Points
  - Hazard Analysis of Process Characteristics
  - Hazard Analysis for Functional Characteristics
  - Examination of Organization & Responsibility
- **Techniques for Functional Characteristics**
  - Software HAZOP
  - Software FTA
  - Software FMEA



Software-Contributable System Hazards

- Software-Contributable System Hazards for DRPS App. S/W

No	Hazard	Criticality Level
1	DRPS cannot generate a trip signal when a trip condition for a process variable is satisfied.	4
2	DRPS generates a trip signal when it should not generate a trip signal.	3
3	DRPS cannot send qualified information of its operating status to the main control room.	2

Criticality Level 4 - The most significant hazard that can drive a plant to an accident,
Criticality Level 3 - A hazard that impacts significantly on the system operation but does not lead to an accident
Criticality Level 2 - A hazard that can affect more or less the system operation
Criticality Level 1 - An insignificant hazard that seldom affects the system availability

Identification of Interface Points

NO	Module	Description
1	Receive_Signal	HYSDLTCH Receive Module
2	PAT_Scheduler	Automatic Test Scheduler
3	Test_Selection	Test Selection Module
4	Trip_Logic	PZR_PR_Hi Trip — Pressurizer Hi Pressure Trip
		SG1_LVL_Lo_RPS Trip — SG-1 Low Level Trip
		SG1_LVL_Lo_ESF Trip — SG-1 Low Level Trip for ESF
		SG1_LVL_Hi Trip — SG-1 Hi Level Trip
		SG1_PR_Lo Trip — SG-1 Low Pressure Trip
		CMT_PR_Hi Trip — Containment Hi Pressure Trip
		CMT_PR_HH Trip — Containment Hi-Hi Pressure Trip
		SG1_FLW_Lo Trip — SG-1 Low Coolant Flow Trip
		PZR_PR_Lo Trip — Pressurizer Low Pressure Trip
		VA_OVR_PWR_Hi Trip — Variable Over Power Hi Trip
		SG2_LVL_Lo_RPS Trip — SG-2 Low Level Trip
		SG2_LVL_Lo_ESF Trip — SG-2 Low Level Trip for ESF
		SG2_LVL_Hi Trip — SG-2 Hi Level Trip
		SG2_PR_Lo Trip — SG-2 Low Pressure Trip
		SG2_FLW_Lo Trip — SG-2 Low Coolant Flow Trip
		LOG_POW_Hi Trip — Log Reactor Power Hi Trip
		DNBR_Lo Trip — Low DNBR Trip
		LPD_Hi Trip — Hi LPD Trip
		CPC_CWP Trip — CPC CWP
5	Test_Results_Handler	Test Results Handling Module
6	HW_MONITORING	Heartbeat Monitoring Module
7	HB_Gen	Heartbeat Generation Module
8	Ch_Bypass_Send_Receive	Channel Bypass Transfer Module
9	Send_Signal	HYSDLTCH Sending Module

- Interface Points between FBD Modules and Hazards
  - Trip modules in no.4 (except CPC_CWP) affect the hazard item 1 and 2.
  - Some S/W in FBD Module no.1 & 2 affect the hazard item 1 and 2 through Trip_Logic (no.4).
  - FBD modules of no.5,8, & 9 affect hazard item 3.

Features of Technical Methods

Software HAZOP	Software FTA	Software FMEA
• All Documents + FBD Code • All S/W Contributable Hazards • Forward, Broad-Thinking Analysis • Brainstorming by HAZOP Members	• Defective S/W Module at Design/Implementation Phase • Most Critical Hazard (CL-4) • Backward, Local Systematic Analysis • Fault Tree by an Individual Analyst	• S/W Code (integrated & tested) • All S/W Contributable Hazards • Forward, Broad Systematic Analysis • By an Individual Analyst
• Deviation Quantity: Qualitative Functional Characteristics • Guide Phrases (Rather Than Guide Words)	• Based on Fault Tree Templates for Function Blocks • Logical Operation in Fault Event	• Based on a Single Failure-Mode Template for All the Function Blocks
• Proven-Technology by NUREG/CR-6430	• Addressed by NUREG/CR-6101 Appendix	• Addressed Partly by NUREG/CR-6101 Appendix
• Need Discussion Skills	• Difficult to Apply to All Scope	•

Conclusions

- For safety critical software, the software V&V and safety analysis were applied to each phase of the software lifecycle.
- Both activities have different perspectives and compensate for each other (one for functional requirements, the other for safety requirements).
- Through the software V&V activities, numerous anomalies were identified at an initial lifecycle and these defects were resolved as the lifecycle proceeded.
- The software safety analysis could find different types of software defects that have not been identified in the V&V works
- Software HAZOP/FTA/FMEA were all very useful, and S/W FMEA is better.
- Further Considerations
  - Cyber Security: How to incorporate it into V&V activities or be a separate process?
  - Quantification of S/W Quality and Reliability

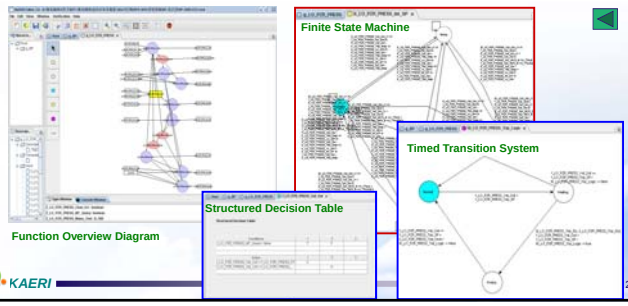
One Further Activity : Quantification

- Bayesian SRGM
  - Purpose: Quantification of Software Code Quality after Testing
    - ✓ identifying the degree of effectiveness of the current test and anticipating the amount of V&V testing in the next phase.
- S/W Failure Probability Estimation
  - Purpose: Reliability of Software
    - ✓ Quantification of V&V Results + Test Results.
- Bayesian Belief Networks
  - Purpose: Quantification of Quality of S/W Development Process
    - ✓ Quantification of All of V&V Activities
- Difficulties in Applying Existing Reliability Model
  - No Sufficient Failure Data to Apply to Existing S/W Reliability Models
    - ✓ Large Amount of Qualitative V&V Activities Revealing S/W Quality
    - ✓ Large Amount of Quantitative Test Activities…

Backup Materials

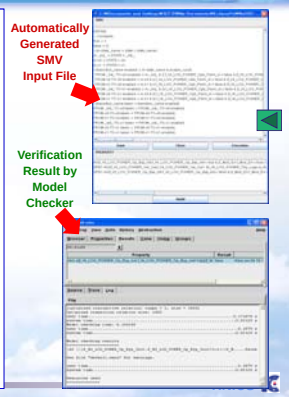
Formal Specification & Verification

- **NuSRS: Formal Requirements Specification/Verification Tool**
  - **Formal Specification Language: NuSCR (A Modified SCR used in NPP)**
  - **NuSCR Language Types: Functional Overview Diagram (FOD), Structured Decision Table (STD), Finite State Machine (FSM), Timed Transition System (TTS)**



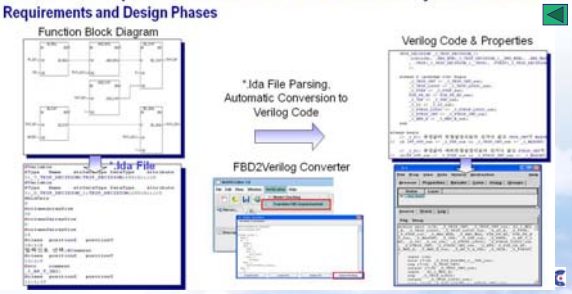
SRS Formal Verification by Model Checking

- **Formal Verification**
  - Automatic Translation of NuSRS from NuSCR into SMV Input Format
  - Model Checking by Cadence SMV incorporated into NuSRS
- **Main Properties to be Verified**
  - ✓ Deadlock Freeness (System can never be in a situation where no progress is possible)
  - ✓ Non-Determinism (System never has some conflicting transitions)
  - ✓ Trip by Error (If module/channel/input error occur, trip signals are fired immediately)
  - ✓ Trip by Logic (Trip signal is fired if the process value rises above the setpoint and this condition lasts for pre-defined time)
  - ✓ Normal Status (If trip conditions are not satisfied, then trip signal shall never be fired)
  - ✓ Trip with OB (Trip signal is never be fired during operating bypass)

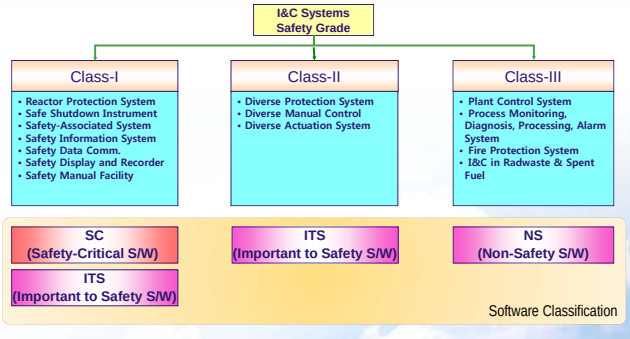


FBD Formal Verifier

- Formal Verification Tool for Design and Implementation Phases
- Automatic Conversion of FBD Program into "Verilog" Language
- Model Checking of Verilog Formal by SMV Model Checker
- The Same CTL Properties as That in NuSRS → Enable Consistency Check between Requirements and Design Phases

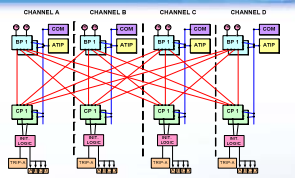


Safety Classification of I&C Systems



<Reference: KINS Reg. Guide, Chap. 8>

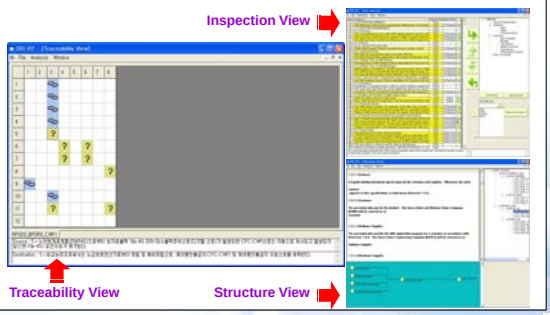
Target S/W: S/W in Digital RPS (IDiPS-RPS)



- **Configuration of IDiPS-RPS (for each channel)**
  - **Two Bistable Processors (BPs):** Determine trip state by signal comparison, SC
  - **Two Coincidence Processors (CPs):** Generate trip signal by a 2/4(2/3) voting, SC
  - **Automatic Test & Interface Processor (ATIP):** Performs Tests(MT/MIAT/PT) & Interfaces with other ATIPs, SR
  - **Cabinet Operator Module (COM):** GUI + H/W (Ch. Bypass, Init. Circuit Reset)
  - **Network Type:** SDL (SC), ICN (SR), ICDN

CASE Tool: SIS-RT

- **SIS-RT: Proprietary Tool for KNICS Project**
- **Three View Modes: Inspection View, Traceability View, Structure View**



Example of Formal Verification

- Accomplishment of Formal Verification
  - All Safety-Critical/Related Software Modules
  - Specification Errors (Vague Initializations, Multiple Test Case Selection), Behavioral Errors (OB)

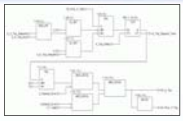
Example of Formal Verification

- Accomplishment of Formal Verification
  - All Safety-Critical/Related Software Modules
  - Specification Errors (Vague Initializations, Multiple Test Case Selection), Behavioral Errors (OB)

Test Coverage Criteria for FBD Program

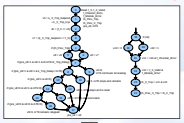
- Incorporate Test Coverage Criteria into Black Box Testing for Compensation of Each Other
- FBD program is converted into a corresponding flow graph and the test coverage criteria is applied to draw necessary test cases.

FBD



Based on Conversion Template

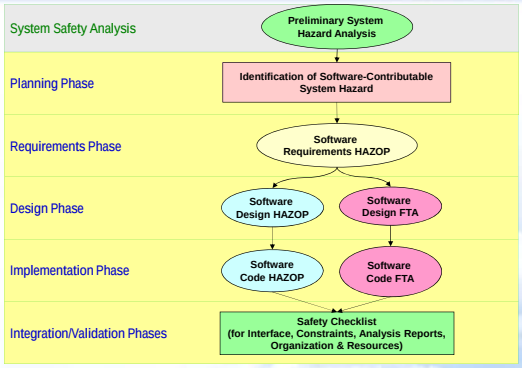
Flow Graph



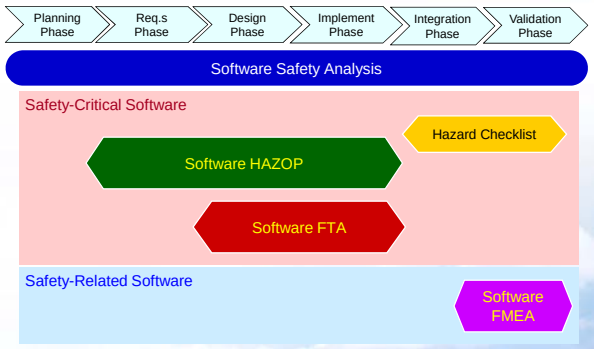
Application of Test Coverage Criteria

Test Case Generation & Test Execution

Software Safety Analysis Activities



Technical Methods of SSA



Analysis Method #1 - Software HAZOP

- **Distinguishing Features of Software HAZOP**
  - **Deviation Quantity: S/W Functional Characteristics (Qualitative)**
    - ✓ **Accuracy, Capacity, Functionality, Reliability, Robustness, Security, Safety**
  - **Guide Phrases rather than Guidewords**

Fun. Charac.	Deviation Checklist	Cause	Analysis	Effect	C	Suggestion
Accuracy	What is the consequence if the sensor value is below its minimum range?	Sensor Failure	The TRIP_DECISION sub-module handles properly an out-of-range value, but it is carried out after all logical operations are done.	No effect on safety, but operability is poor.	2	It is desirable that a trip signal occurs at the front when an out-of-range sensor input value exists.
Accuracy	What is the consequence if the sensor value is above its maximum range?	Sensor Failure				
Accuracy	What is the consequence if the sensor value is within its physical range but incorrect?	Sensor Failure or, Input Conditioner Malfunction	This is the problem at input conditioning processor.	Severe effect on safety	4	Measures should be provided at input processor.
Accuracy	What is the consequence if the internal constant is given a wrong value?	Wong constant value allocation	If MAXCNT is set to 0, the trip signal is always ON regardless of the trip condition status.	Poor Operability	3	Need careful attention when assigning a value.
			If MAXCNT is too large, the trip signal is generated at much later time.	Violating the system response time	4	
Capacity	What is the consequence when an unexpected input signal is arrived?	ATIP Error	No part performs an exceptional handling when ATIP sets up an erroneous test operation.	Wrong test execution	1	Augment test mode selection.
Functionality	What is the consequence if some portions in the FBD module have a defect or cannot perform its intended behavior?	Error in Logic Operation	Pretrip is cancelled whenever it is triggered at the pretrip sub-module.	Pretrip is never functioning	3	Modify a pretrip logic.
			The hysterisis is not reflected in the trip logic sub-module because of using 19th previous value.	Inducing a trip malfunction	4	Modify trip logic.

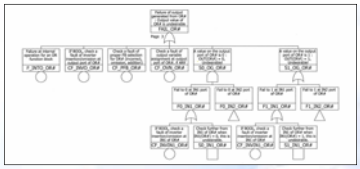
Analysis Method #1 - Software HAZOP

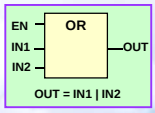
- **Distinguishing Features of Software HAZOP**
  - **Deviation Quantity: S/W Functional Characteristics (Qualitative)**
    - ✓ **Accuracy, Capacity, Functionality, Reliability, Robustness, Security, Safety**
  - **Guide Phrases rather than Guidewords**

Fun. Charac.	Deviation Checklist	Cause	Analysis	Effect	C	Suggestion
Accuracy	What is the consequence if the sensor value is below its minimum range?	Sensor Failure	The TRIP_DECISION sub-module handles properly an out-of-range value, but it is carried out after all logical operations are done.	No effect on safety, but operability is poor.	2	It is desirable that a trip signal occurs at the front when an out-of-range sensor input value exists.
Accuracy	What is the consequence if the sensor value is above its maximum range?	Sensor Failure				
Accuracy	What is the consequence if the sensor value is within its physical range but incorrect?	Sensor Failure or, Input Conditioner Malfunction	This is the problem at input conditioning processor.	Severe effect on safety	4	Measures should be provided at input processor.
Accuracy	What is the consequence if the internal constant is given a wrong value?	Wong constant value allocation	If MAXCNT is set to 0, the trip signal is always ON regardless of the trip condition status.	Poor Operability	3	Need careful attention when assigning a value.
			If MAXCNT is too large, the trip signal is generated at much later time.	Violating the system response time	4	
Capacity	What is the consequence when an unexpected input signal is arrived?	ATIP Error	No part performs an exceptional handling when ATIP sets up an erroneous test operation.	Wrong test execution	1	Augment test mode selection.
Functionality	What is the consequence if some portions in the FBD module have a defect or cannot perform its intended behavior?	Error in Logic Operation	Pretrip is cancelled whenever it is triggered at the pretrip sub-module.	Pretrip is never functioning	3	Modify a pretrip logic.
			The hysterisis is not reflected in the trip logic sub-module because of using 19th previous value.	Inducing a trip malfunction	4	Modify trip logic.

Analysis Method #2 - Software FTA

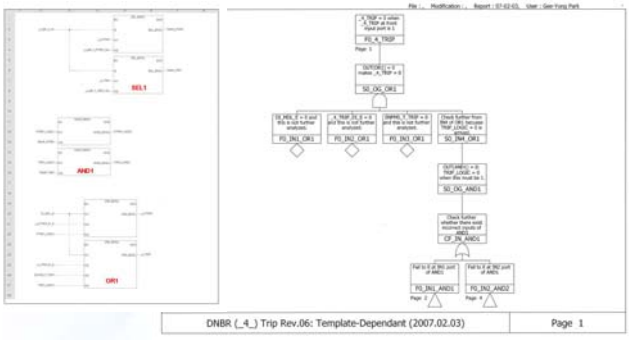
- Software FTA was applied to the critical FBD modules identified by the software HAZOP.
- Software FTA was constructed based on the fault-tree templates.
- There is a variety of fault-tree templates: Each function block in FBD program has its own fault-tree template.
- Software FTA provides a systematic way to find a local and logical software defect that is hard to find by S/W HAZOP or V&V works.





Fault Tree Template for OR Function Block

Analysis Method #2 - Software FTA Result #1

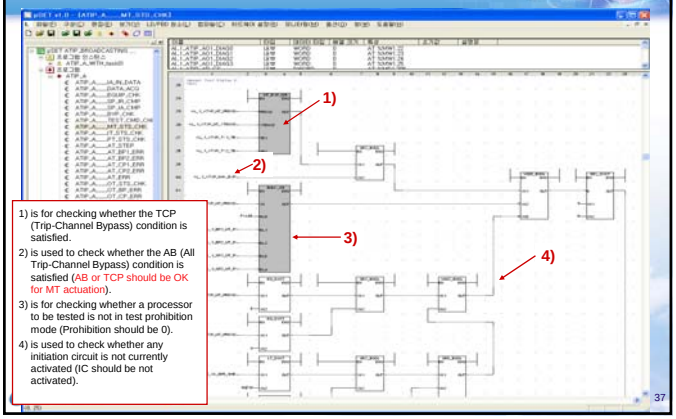


Analysis Method #3 - Software FMEA

- Software FMEA was performed based on the failure-mode template.
- There is a single failure-mode template that are equally applicable to all the function blocks of the FBD program code.
- Software FMEA provides a very systematic and efficient way to find a software defect that had been passed through tests.

ITEM	FAILURE MODES	DESCRIPTION
Function	Omission	- Check the function omitted from a module
	Incorrect Realization	- Check the correctness of a program module - Check the selection of a function block(FB) is correct (e.g., LE FB is replaced by LT FB) - Check the arithmetic operation is correct (e.g., division by zero)
	Unintended Addition	- Check the function added to a program module - If any, check this addition is appropriate with respect to overall safety function or high-level requirements
	Function Interaction	- Check the input condition for a function is correct - Check the output condition (of a control signal) is correct and it has a correct effect on neighboring functions, after function is completed
Input	Input Definition	- Check the input port has a correct name - Check the input variable is assigned a correct address - Check the type of input variable and the consistency between this input and the output of a FB executed previously
	Input Value	- Check the input value/constant is correct - Check the input value is within the accepted range - Check the initial value assignment of an input value
	Input Timing	- Check the input value is obtained at a correct time or by a correct sequence - Check the function and the effect of a timer, if any
	Input Format	- Check the "Inverter" is used correctly - Check input swapping or input disorder - Check the "EN" connection - Check the input number of an FB is appropriate
Output	Output Definition	- Check the output port of an FB has a correct output variable - Check the external output variable is assigned to a correct address - Check the type of output variable
	Output Value	- Check the output value is correct to a receiving object - Check the initial assignment of an output variable - Check the output value is within a predefined, meaningful range
	Output Timing	- Check the output generated from an execution period or at a location in the program module is appropriate - Check the output mechanism of a timer is correct and the effect of it is appropriate
	Output Format	- Check the "Inverter" at the output port - Check output swapping or output disorder - Check the output number of an FB is appropriate

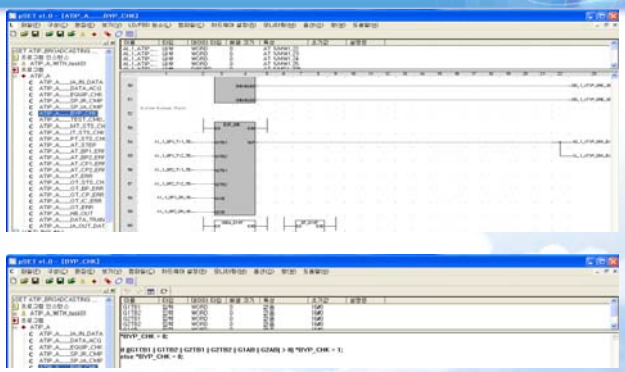
MT_STS_CHK Module for MT



1) is for checking whether the TCP (Trip-Channel Bypass) condition is satisfied.
2) is used to check whether the AB (All Trip-Channel Bypass) condition is satisfied (AB or TCP should be OK for MT actuation).
3) is for checking whether a processor to be tested is not in test prohibition mode (Prohibition should be 0).
4) is used to check whether any initiation circuit is not currently activated (IC should be not activated).

37

"BYP_CHK" in BYP_CHK Module



KAERI KNICS 38

Result of A Defect in MT_STS_CHK

❑ **Consequences of a Defect in MT_STS_CHK Module**
- ✓ When any other channel (say, Channel D) is already all bypassed, Channel A can execute the manual test even though its trip channel is not bypassed.

❑ **Difficulty in Finding Out This Defect through the System Test**
- ✓ Four fully-interconnected RPS channels required.
- ✓ More than at least 2 test persons.
- ✓ Complex Test Procedures
  - ➢ Variables to be tested: 18 trip variables per channel
  - ➢ Channels to be tested: 4 channels
  - ➢ Test Type: Manual Test, Manual-Initiated Automatic Test

KAERI KNICS 39