IPSN

DES--172 f.

RAPPORT

RAPPORT DES/172 f

CONTRIBUTION A L'EVALUATION DE SURETE DES LOGICIELS DE CONTROLE-COMMANDE DES CENTRALES NUCLEAIRES : APPLICATION AU SPIN N4

SOUBIES B., BOULC'H J., ELSENSOHN O.,
LE MEUR M., HENRY J.Y.

cea INSTITUT DE PROTECTION ET DE SURETE NUCLEAIRE
DEPARTEMENT D'EVALUATION DE SURETE

RAPPORT DES/172 f

CONTRIBUTION A L'EVALUATION DE SURETE DES LOGICIELS DE CONTROLE-COMMANDE DES CENTRALES NUCLEAIRES : APPLICATION AU SPIN N4

SOUBIES B., BOULC'H J., ELSENSOHN O.,
LE MEUR M., HENRY J.Y.
(IPSN/DES/SAMS)

Journée
"Sûreté et contrôle-commande
des centrales nucléaires"
SFEN
PARIS, 22 mars 1994

Juin 1994

E
E

SOMMAIRE

RESUME

Le processus d'autorisation de fonctionnement des centrales nucléaires comprend des étapes obligatoires qui donnent lieu notamment à un examen détaillé du contrôle-commande.

Cet examen prend en compte les aspects liés aux technologies (circuits intégrés, logiciels) qui ont été choisies par le fabricant pour les systèmes programmés qui assurent des fonctions classées de sûreté.

Des innovations importantes ont été introduites par l'exploitant dans le procédé de conception et de réalisation des systèmes de sûreté des réacteurs à eau sous pression de 1400 MWe et, en particulier, pour le système de protection intégré numérique (SPIN).

La méthodologie appliquée par l'IPSN pour l'examen des logiciels de ce système est présentée dans cette communication. Cette méthodologie conduit l'IPSN à mener des études et des développements d'outils dont l'objectif premier est d'apporter une aide à l'analyse.

1. Introduction

Les réacteurs à eau sous pression (REP) de 1300 MWe, comme ceux de 1400 MWe, fonctionnent avec des systèmes de sûreté dont la technologie est à base de microprocesseurs. C'est en particulier le cas pour le Système de Protection Intégré Numérique (SPIN) qui assure le déclenchement de l'arrêt d'urgence et la majorité des fonctions de sauvegarde. Les logiciels utilisés dans ces systèmes doivent donc présenter un haut niveau de confiance dans la réalisation de leurs fonctions. Dans le cas du SPIN, trois acteurs agissent à des degrés différents pour atteindre cet objectif. Il s'agit :

- du fabricant du système de protection, MERLIN GERIN,
- du concepteur de la chaudière nucléaire, FRAMATOME,
- de l'exploitant des centrales nucléaires, ELECTRICITE DE FRANCE (EDF), responsable de la sûreté de ses installations.

Les autorisations réglementaires sont délivrées par l'Autorité de Sûreté française (Direction de la Sûreté des Installations Nucléaires), après examen des dispositions techniques retenues par l'exploitant. Cet examen est réalisé par l'Institut de Protection et de Sûreté Nucléaire (IPSN) et le groupe permanent chargé des réacteurs nucléaires.

Cette communication présente :

- les méthodes utilisées par le fabricant pour le développement des logiciels du SPIN des REP de 1400 MWe (palier N4),
- la démarche suivie par l'IPSN pour l'évaluation des logiciels de sûreté du système de protection du palier N4.

2. Méthodes utilisées par le fabricant pour le développement des logiciels du SPIN

2.1. Présentation du SPIN

Le système de protection des REP de 1400 MWe, comme celui des REP de 1300 MWe, comprend le Système de Protection Intégré Numérique (SPIN) qui est constitué de:

- quatre Unités d'Acquisition et de Traitement de Protection (UATP) redondantes et indépendantes,
- deux Unités Logiques de Sauvegarde (ULS) redondantes et indépendantes.

Ces unités sollicitent l'ouverture des disjoncteurs d'arrêt d'urgence et commandent les actionneurs de sauvegarde lorsque deux mesures redondantes parmi les quatre d'un même paramètre physique ont franchi un seuil prédéterminé.

Dans le cas du palier N4, chaque unité du SPIN est constituée principalement d'un microprocesseur MOTOROLA 68000. Son logiciel, dont le code binaire est implanté en REPROM, est développé en langage C et en assembleur 68000. Les échanges d'information entre les unités du SPIN se font par l'intermédiaire de réseaux locaux :

- huit réseaux locaux de protection, de type NERVIA, redondants assurant notamment les échanges entre les UATP et les ULS,
- deux réseaux locaux de signalisation, de type NERVIA, redondants,
- des réseaux actionneurs internes aux ULS assurant la transmission des ordres de protection entre les unités de traitement et les cartes actionneurs.

Des unités spécifiques pour les tests périodiques sont intégrées dans le SPIN.

2.2. Développement des logiciels du SPIN

La démarche générale de ces développements est décrite dans le Plan Qualité Logiciel du fabricant MERLIN GERIN. Le développement du logiciel s'appuie principalement sur un atelier de Spécification Assistée et de Génération d'Application (SAGA), sur des règles de programmation, sur la séparation des équipes de conception et de vérification.

Le cycle de développement d'un logiciel dans ce contexte comprend sept phases dont chacune donne lieu à un ou plusieurs documents. Ces phases et les documents associés sont :

- la spécification du logiciel à laquelle est associé le Cahier des Charges Logiciel (CCL),
- la conception préliminaire à laquelle est associé le dossier de Conception Préliminaire Logiciel (CPL),
- la conception détaillée à laquelle est associé le dossier de Conception Détaillée Logiciel (CDL),
- le codage auquel sont associés les dossiers de programmation concernant les listages des instructions des différents composants constituant le logiciel,
- les tests unitaires auxquels sont associés les dossiers de tests de chaque composant,
- les tests d'intégration du logiciel auxquels sont associés les dossiers de Tests d'Intégration Logiciel (TIL),
- les tests de validation du logiciel auxquels sont associés les dossiers de Tests de Validation Logiciel (TVL).

Quatre de ces documents (CCL, CPL, CDL, TVL) donnent lieu, chacun, à une revue entre les responsables des équipes de conception et de vérification et le responsable de l'assurance qualité. Ces revues font partie des actions "qualité" qui interviennent au cours du développement.

De plus, le CCL est soumis à l'approbation du concepteur de la chaudière nucléaire.

Après chaque revue satisfaisante, la phase suivante du développement du logiciel est entreprise.

L'atelier SAGA participe directement à plusieurs phases du cycle de développement. Il met en oeuvre cinq outils :

- l'outil de spécification,
- l'outil de génération de code,
- l'outil de programmation,
- l'outil de documentation,
- l'outil d'administration.

L'outil de spécification intervient lors des phases de conception préliminaire ou détaillée. Par son interface interactive et graphique, il permet la description descendante du logiciel à développer en composants qui sont, à leur tour, détaillés jusqu'à obtenir des composants simples à programmer.

L'outil de génération de code permet d'obtenir le programme en langage C du logiciel spécifié précédemment.

L'outil de programmation aide le programmeur lors de la conception et de l'élaboration du programme source d'un composant, non généré automatiquement avec l'outil précédent, en lui proposant un canevas type et en contrôlant le respect de certaines règles d'écriture. L'interface de ce composant est générée automatiquement à partir de sa conception.

L'outil de documentation intervient au fur et à mesure des phases de conceptions préliminaire et détaillée, et met en forme la documentation textuelle associée aux composants décrits au moyen de l'outil de spécification.

L'outil d'administration permet la gestion de l'accès aux ressources de l'atelier par les différents utilisateurs.

Les règles de programmation doivent permettre de respecter particulièrement les prescriptions de la norme CEI 880 intitulée "Logiciel pour les calculateurs dans les systèmes de sûreté des centrales nucléaires", et d'imposer une programmation homogène facilitant le test et la maintenance des logiciels.

La séparation des équipes de conception et de vérification, déjà utilisée dans le plan qualité logiciel des REP 1300 MWe, a été retenue pour accroître les contrôles indépendants.

3. Evaluation des logiciels du SPIN par l'exploitant nucléaire

La démarche de sûreté retenue par l'exploitant consiste à faire la démonstration du respect des fonctions de sûreté par l'étude de scénarios accidentels requérant des matériels ou des systèmes classés qui doivent répondre aux exigences de conception, de fabrication et d'installation.

S'agissant du système de protection du réacteur, l'exploitant effectue, pour les logiciels de sûreté du SPIN, en complément aux dispositions prises dans le Plan d'Assurance de la Qualité du fabricant, une validation indépendante. Il en approuve les spécifications et pratique des audits notamment lors des tests réalisés en phase de validation logiciel chez le fabricant.

Par ailleurs, les tests réalisés sur chaque équipement du SPIN (UATP et ULS) pourvu des logiciels validés, puis sur le SPIN et ses interfaces avec les autres systèmes, donnent lieu à des revues communes entre le fabricant, le concepteur de la chaudière et l'exploitant.

4. Méthodologie d'évaluation des logiciels de sûreté utilisée par l'IPSN

L'appui technique (IPSN-DES-SAMS) de l'autorité de sûreté (DSIN) a pour charge de réaliser toutes les investigations qu'il juge nécessaires afin de s'assurer que les méthodes et techniques mises en oeuvre par le fabricant et l'exploitant garantissent, pour les logiciels du SPIN, la sûreté attendue et permettent une testabilité et une maintenabilité suffisantes. Pour ce faire il porte plus particulièrement son attention sur les aspects suivants:

- méthodes de développement des logiciels rationnelles et rigoureuses suivant un plan précis d'assurance de la qualité (documentation et code);

- règles strictes de programmation pour la production d'un logiciel testable et maintenable (code);
- tests mis en oeuvre pour assurer un taux de couverture suffisant aussi bien chez le fabricant que sur site (simulation).

L'évaluation menée par l'IPSN ne porte pas sur la totalité des équipements constitutifs du SPIN, vu leurs complexités. Il a été décidé d'effectuer l'analyse sur:
- la totalité de la documentation associée aux spécifications techniques du SPIN,
- un ensemble représentatif des fonctions du système de protection.

L'ensemble représentatif, choisi pour l'évaluation de sûreté, comporte deux chaînes, l'une relative à une demande d'arrêt d'urgence et l'autre à une demande d'injection de sécurité. Elles mettent en jeu chacune des unités fonctionnelles nécessaires à l'accomplissement d'une tâche de sûreté :
- deux unités d'acquisition des données du procédé,
- une unité de traitement de ces données permettant l'élaboration d'un déclenchement partiel correspondant à une demande d'arrêt d'urgence ou à une demande d'action de sauvegarde,
- une unité chargée du vote majoritaire qui commande les interrupteurs d'arrêt d'urgence et les actions de sauvegarde.

Les logiciels associés à cet ensemble représentatif traitent une ou plusieurs données issues du procédé, depuis l'acquisition jusqu'à l'élaboration d'un ordre. La méthodologie adoptée pour analyser les logiciels du SPIN N4 procède par étapes successives pour évaluer les différentes solutions techniques proposées par l'exploitant. Actuellement, les différentes étapes mises en oeuvre lors de l'évaluation des logiciels de sûreté sont au nombre de six et sont données ci-dessous:
- étape 1, examen critique des documents (Cf. paragraphe 4.1),
- étape 2, évaluation de la qualité du code (Cf. paragraphe 4.2),
- étape 3, détermination des composants logiciels critiques (Cf. paragraphe 4.3),
- étape 4, élaboration des jeux d'essais (Cf. paragraphe 4.4),
- étape 5, étude de conformité (Cf. paragraphe 4.5),
- étape 6, étude de robustesse (Cf. paragraphe 4.6).

Certaines de ces étapes sont menées en parallèle, c'est le cas des étapes 2, 3 et 5. Les étapes 1 et 2 sont plus spécifiquement orientées vers une analyse dite statique, car elles ne nécessitent pas l'exécution du programme. Les étapes 4, 5 et 6 sont, elles, tournées vers une analyse dite dynamique. L'étape 3 constitue la transition entre les étapes d'analyse statique et les étapes d'analyse dynamique. Pour permettre une démarche acceptable en termes de tâches à réaliser et pour apporter des éléments techniques à l'analyste, des outils d'aide ont été développés. Il s'agit de:
- l'outil de modélisation des textes en langage naturel qui permet d'évaluer la complétude et la cohérence des documents de spécification et de conception,
- l'outil d'analyse statique de programme qui permet d'évaluer la qualité de la programmation et offre, grâce à son analyseur sémantique, une aide à la génération de jeux d'essais,
- l'outil permettant de déterminer les composants critiques en fonction des objectifs de sûreté que le logiciel doit assurer,
- l'atelier de simulation qui est composé des outils suivants:

* l'outil de simulation et de test permettant de réaliser des analyses dynamiques;
* l'outil de description des programmes d'environnement pour la simulation;
* l'outil de traitement des résultats qui présente de façon graphique les résultats des analyses dynamiques.

L'IPSN, par les analyses décrites dans les paragraphes suivants, pourra réaliser les objectifs précisés plus haut et répondre ainsi aux attentes de l'autorité de sûreté.

4.1. Examen critique des documents

L'évaluation de sûreté des systèmes programmés conduit l'IPSN à porter un jugement sur la pertinence des informations contenues dans les documents de spécification et de conception des logiciels, par rapport à la connaissance technique du projet et aux normes appliquées à l'installation qui utilise ces systèmes.

L'examen du système de protection effectué dans le cadre de cette évaluation, prend en compte les exigences de sûreté de l'installation, l'architecture du système et ses spécifications. Il s'agit alors de vérifier la présence de toutes les fonctions requises pour assurer la sûreté de l'installation et le respect de la diversité fonctionnelle qui permet de se protéger des défauts de mode commun.

Ces fonctions utilisent des signaux de protection, par exemple le haut niveau dans les générateurs de vapeur, et conduisent à des actions de protection (arrêt d'urgence ou sauvegarde). Plusieurs signaux de protection apparaissant lors d'une même séquence accidentelle doivent être traités dans des unités fonctionnelles différentes du SPIN (principe de la diversité fonctionnelle). C'est par exemple le cas des signaux de très basse pression dans le pressuriseur et de très haute pression dans l'enceinte de confinement qui apparaissent lors d'un accident de perte de réfrigérant primaire (APRP grosses brèches) et qui sont traités dans deux unités fonctionnelles différentes du SPIN.

Le développement des logiciels correspondant aux fonctions du système de protection est organisé par un Plan d'Assurance Qualité Logiciel (PAQL) qui donne lieu à l'élaboration d'une documentation volumineuse, tout au long du cycle de développement.

Cette dernière comprend des documents écrits en langage naturel (spécifications, conception, tests) et le programme lui-même.

L'ensemble de la documentation est produit sur une période de temps importante, du fait de l'ampleur de ce type de logiciel.

Chaque document est analysé non seulement pour comprendre les fonctions réalisées, mais surtout pour vérifier s'il existe des informations superflues (sources de complexité), manquantes ou incohérentes en regard de l'ensemble de la documentation concernant ces logiciels.

La méthode AVIS et son support informatique AVISO aident à réaliser cet examen par une démarche systématique et rigoureuse.

La méthode s'appuie sur l'analyse linguistique pour constituer des graphes qui représentent les informations décrites dans le document.

Cette opération met en relation chaque élément du texte avec la partie correspondante du graphe.

Par rapport à un langage discursif ou mathématique, la modélisation ainsi opérée présente l'avantage de mettre en évidence les relations entre les informations portées par le texte. La vision globale qui en découle permet de focaliser l'attention aussi bien sur le sens que sur les détails associés à chaque information.

Cette représentation facilite l'examen de la cohérence et de la complétude de ces informations.

En outre, les références établies entre le texte et le graphisme permettent de lier au texte originel les anomalies relevées lors de la modélisation.

Ainsi, l'analyse d'une documentation produite sur une période longue et modifiée à chaque nouvelle version de logiciel est facilitée par la capacité de cet outil à mémoriser les informations contenues dans les différents textes avec les commentaires et les remarques soulevés par l'analyse.

4.2. Evaluation de la qualité du code

Les logiciels de l'ensemble représentatif sont analysés afin de :

- rechercher des constructions dangereuses pour le type de langage utilisé :
  - . anomalies du flux de données (définitions et utilisations des valeurs des variables, type des variables...),
  - . expressions arithmétiques (parenthèses, division par zéro...),
- rechercher une structuration non correcte ou trop complexe des programmes :
  - . boucles à entrées ou sorties multiples,
  - . boucles à indices variables,
  - . code inaccessible,
  - . code inutile,
- vérifier le respect des prescriptions de la CEI 880 jugées importantes par l' IPSN.

Les composants dans lesquels ont été détectées des anomalies de programmation deviennent des composants dits sensibles. La testabilité et la maintenabilité de ces composants sont à leur tour évaluées. Certains de ces composants, essentiellement ceux qui comportent des boucles à indice variable, pourront donner lieu à des tests, lors de l'étude relative à la robustesse du logiciel, permettant de vérifier leur comportement dans ces cas là.

Une première série d'analyses a été effectuée au moyen d'un outil d'analyse statique de programme (analyseurs structurels de l'outil MALPAS) sur le logiciel d'une des unités fonctionnelles du SPIN. L'exploitation des résultats a permis de déceler quelques particularités de programmation pouvant influencer la testabilité et la maintenabilité de ce logiciel.

4.3. Détermination des composants logiciels critiques

Les logiciels des unités choisies traitent de plusieurs chaînes. Il faut donc identifier les parties relatives aux deux chaînes sélectionnées de l'ensemble représentatif, que l'on appelle composants essentiels.

Parmi ces composants, on recherche les fonctions dites critiques dont une défaillance est susceptible d'entraîner un dysfonctionnement grave du système.

Cette recherche s'effectue au moyen de la Méthode d'Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité (AMDEC) adaptée à l'analyse des logiciels. Elle constitue la première étape de la démarche AFFUT qui a pour objectif de déterminer les fonctions unitaires les plus importantes à soumettre aux tests réalisés par l'IPSN.

Celle-ci consiste, après une étude fonctionnelle, à évaluer les effets des défaillances postulées tour à tour sur chaque fonction des logiciels.

Il est alors possible de calculer un indice de criticité pour chaque fonction, en prenant en compte le nombre et la gravité des défaillances, et ainsi d'en établir le classement.

La seconde étape de la démarche AFFUT consiste à étudier les fonctions critiques de façon détaillée en analysant l'ensemble des tests effectués par le fabricant.

Si ces tests sont insuffisants pour s'assurer que les défaillances postulées ne peuvent se produire, ces fonctions critiques réalisées par des composants qui sont appelés à leur tour critique feront l'objet de tests complémentaires lors de l'étude de consistance et de l'étude de robustesse.

4.4. Elaboration des jeux d'essais

Cette phase est constituée de deux parties. La première est basée sur un examen des tests du fabricant en vue de l'étude de la consistance, la deuxième s'appuie, actuellement, sur les résultats de l'analyseur sémantique de l'outil MALPAS et est orientée vers l'étude de la robustesse.

Dans la première partie, l'analyste choisit parmi les jeux de tests du fabricant ceux qui correspondent à des conditions spécifiées de fonctionnement du système pour vérifier le comportement de celui-ci.

Dans la deuxième partie, l'étude des composants critiques et sensibles que l'analyste a retenus se poursuit au moyen de l'outil PEGASE. Celui-ci permet de donner tous les chemins fonctionnels conduisant aux différentes valeurs que peut prendre chaque variable de sortie du composant considéré. Il permet de trouver les plages de valeurs des données d'entrée à partir des relations conditionnelles décrivant les chemins fonctionnels.

Des valeurs sont choisies pour les données d'entrée afin d'activer les composants critiques lors des tests.

Cette analyse permet aussi de vérifier les plages de variation des données par rapport à leur spécification.

En outre, ce type d'analyse montre les dépendances entre les données d'entrée et les données de sortie, et rend possible la vérification de la conformité entre le code et la spécification du logiciel, si cette dernière comporte de telles informations.

4.5. Etude de consistance

L'évaluation des logiciels de l'ensemble représentatif donne lieu à une analyse dynamique qui permet de déterminer, dans une première phase, la consistance du comportement de ces logiciels au regard de leurs spécifications.

L'étude de consistance permet de vérifier, sur l'exemple représentatif cité précédemment, les valeurs prises par les sorties de ces chaînes (par exemple la commande d'un arrêt d'urgence) lorsque les entrées prennent des valeurs choisies par l'analyste dans le domaine nominal de fonctionnement du système de protection. Cette étude vérifie les aspects les plus significatifs du comportement du programme binaire qui est réellement utilisé sur site.

L'IPSN a développé, pour ce type d'examen , un ensemble d'outils qui permet de réaliser la simulation du fonctionnement par le déroulement du programme binaire sans avoir besoin du matériel (carte de l'unité centrale, cartes périphériques...) qui est utilisé sur site. Ces outils, qui utilisent un ordinateur, permettent :

- la constitution d'un environnement qui reproduit les échanges entre chaque microprocesseur et les circuits (horloges, circuits de communication, mémoires...) qui lui sont associés dans chaque unité du système de protection installé sur site,
- l'exécution des programmes binaires des unités du système de protection par un simulateur de microprocesseurs, avec la production de fichiers spécifiques qui tracent toutes les interactions entre les microprocesseurs et leurs environnements, avec mention du temps d'exécution,
- la présentation, sous une forme synthétique (chronogrammes, courbes...), des valeurs prises par les différentes variables surveillées, afin de permettre une analyse des résultats de simulation.

La reconstitution de l'environnement du programme binaire et du microprocesseur qui l'exécute est obtenue par le développement de logiciels spécifiques qui remplacent les matériels sollicités par ces programmes. Ce développement est fait en utilisant essentiellement une description graphique basée sur la méthode SADT.

L'exécution des programmes tient compte des valeurs des variables d'entrée données par les jeux d'essais conçus pour cette étude de consistance.

La mise en place d'un tel système simulé est actuellement en cours. Dans un premier temps, une sélection des conditions de fonctionnement normales du système de protection sera exécutée pour s'assurer de l'adéquation de la modélisation obtenue par l'environnement développé pour cette étude. Dans un second temps, des exécutions seront réalisées pour vérifier le comportement des logiciels du système lorsque celui-ci est mis en situation de fonctionnement particulière (dégradation de la logique de vote 2/4, par exemple) prévues dans la spécification.

Le système simulé et ses jeux d'essais seront réutilisés pour vérifier la non-régression du bon fonctionnement de chaque version de ces logiciels.

4.6. Etude de robustesse

Cette étude a pour objectif principal de juger du comportement des logiciels de l'ensemble représentatif soumis à des jeux d'essais, définis auparavant, qui représentent des dysfontionnements du système de protection ou des systèmes lui délivrant des informations. Les jeux d'essais sont focalisés sur les composants critiques ou sensibles détectés lors des étapes précédentes. Elle met en place une analyse qui présente un aspect complémentaire aux tests réalisés par le fabricant.

Cette étude utilise les outils de simulation décrits pour l'étude de consistance, afin de constituer un environnement plus complet permettant notamment d'atteindre certaines variables internes des logiciels qui sont représentatives des dysfonctionnements recherchés.

Les résultats des simulations obtenus avec les différents jeux d'essais pour la robustesse doivent être analysés pour identifier l'état de chaque variable de sortie de l'ensemble représentatif du système. Ceci nécessite de connaître les valeurs qui devraient être obtenues pour chaque cas de test. Des analyses sémantiques spécifiques sont réalisées pour calculer les valeurs attendues (oracle).

L'analyse des résultats de simulation est conduite pour identifier, au niveau des sorties du système, les conséquences des dysfonctionnements introduits et d'en tirer les conclusions sur l'adéquation des comportements du système au regard des missions qu'il doit assurer.

5. Conclusion

Les outils décrits précédemment sont opérationnels ou en cours d'expérimentation et représentent un choix sur la base des techniques connues actuellement. Des évolutions dans ce domaine apparaissent en permanence et peuvent conduire à changer les moyens mis en œuvre pour mener une ou plusieurs de ces analyses. L'IPSN consacre une part importante de ses efforts à développer des programmes d'études sur ces sujets.

Cependant, la méthodologie d'évaluation exposée dans les paragraphes précédents peut être considérée comme une base satisfaisante pour l'examen des différents aspects des logiciels de sûreté. Cette démarche est évolutive, d'autres voies restent à explorer. Elles portent, entre autres, sur les autotests inclus dans les logiciels des équipements du système de protection dont l'exhaustivité influence la sûreté de fonctionnement de ce système. De la même façon, le problème posé par les modes communs logiciels et l'extension de cette méthode d'évaluation à d'autres types de systèmes "temps réel" seront approfondis dans un futur proche.

BIBLIOGRAPHIE

BUSSAC J.P.[1], JOVER P.[2], CONFLANT M.[2]
" The Introduction of Computer Systems into Nuclear Power Plant Instrumentation and Control: the French Safety Approach"
International Symposium on Nuclear Power Plant Instrumentation and Control
TOKYO, JAPAN Mai 1992

SOUBIES B., LE MEUR M., HENRY J.-Y., BOULC'H J. [1]
"Evaluation Methods for the Instrumentation and Control Safety Softwares of Nuclear Power Plants"
AIEA Specialists' Meeting on Software Engineering in Nuclear Power Plants : Experience, Issues and Directions
CHALK RIVER, Ontario,CANADA Septembre 1992

Notes :
1 Institut de Protection et de Sûreté Nucléaire CEA CENTAR
2 Direction de la Sûreté des Installations Nucléaires DSIN CENFAR