Un système d'analyse de la qualité: de la norme au produit en passant par le raffinement

Dominique Cansell, Dominique Méry, Cyril Proch

Un système d'analyse de la qualité: de la norme au produit en passant par le raffinement

Dominique Cansell, Dominique Méry, Cyril Proch

LORIA, BP 239 Campus Scientifique, 54506 Vandœuvre-lès-Nancy

Résumé:

Le projet RNRT EQUAST a pour but la réalisation d'un outil de mesure de la qualité de service en télévision numérique terrestre (TNT). Une norme (Digital Video Broadcasting DVB; Measurement guidelines for DVB systems. ETSI TR 101 290 v1.2.1) identifie un certain nombre de contrôles et de paramètres permettant l'évaluation de la qualité de transmission du réseau. La mise en oeuvre de cette norme en un outil implique des calculs et des contraintes temps-réel fortes; elle nécessite une modélisation préalable du système constitué par les paramètres de ladite norme. A partir des documents de normalisation et en relation avec nos partenaires, nous avons extrait et conçu des modèles B événementiels intégrant progressivement, par la relation de raffinement, tous les paramètres à évaluer. Le raffinement assure la cohérence par la preuve du modèle final obtenu et apporte une hiérarchie de dépendances entre les paramètres de la norme. Cette hiérarchie est produite à partir de l'invariant du modèle du système produit et permet de proposer une architecture pour la conception de l'outil de mesure. Ainsi, nous pouvons proposer un ordonnancement correct des tâches de l'application. La connaissance de cet ordonnancement ainsi que la vue structurée du système aide le concepteur dans ses choix d'implantation électronique. Les modèles abstraits du système sont utilisés d'une part pour la mise en évidence de l'organisation des traitements attachés aux paramètres et d'autre part pour la traduction dans un ensemble de programmes SystemC conservant les propriétés des modèles. Afin de demeurer dans une approche préservant les propriétés, nous avons dû modéliser le scheduler SystemC décrit dans le manuel SystemC et monter que les traductions automatisées préservaient effectivement les propriétés des modèles abstraits dans les programmes SystemC.

Mots-clés: méthode B événementielle, raffinement, modélisation formelle, preuve de système, système sur puces.

1.Introduction

La métrologie vise à évaluer la qualité de transmission d'un signal (audio, vidéo, ...) ; ce problème revient à déterminer des critères liés à l'évaluation de la qualité de service, appelée QdS. Le choix d'un critère plutôt qu'un autre est souvent empirique et la mesure de qualité obtenue est souvent subjective. Les éventuelles interactions des différents critères et la vision globale du système sont rarement connus et pris en compte.

1.1 La télévision numérique

Le télévision numérique terrestre DVB-T (Digital Video Broadcasting Television) a introduit des normes afin d'évaluer la qualité de transmission des programmes. En DVB-T, tous les programmes sont émis sur le même média, constituant un flux. Le flux est constitué de différents paquets qui contiennent les informations nécessaires à la reconstruction (données, gestion) des programmes. Une première norme [10] concerne l'évaluation du transit du signal à l'aide d'un certain nombre de paramètres. Ce premier jeu de paramètres d'évaluation se bornait à vérifier la cohérence des informations, le bon respect des structures et la bonne construction de l'entête des paquets.

L'insuffisance de cette première norme est apparue assez vite et une nouvelle norme est apparue. Cette nouvelle norme [9], construite sur la norme précédente, ajoute de nouveaux paramètres, ainsi qu'un certain nombre de paramètres dits de Qualité de Service (QdS). Ces paramètres QdS sont des compositions de paramètres déjà présents considérés comme importants.

Les nouveaux paramètres QdS ajoutés sont issus d'une étude psycho-visuelle de grande envergure menée par TDF C2R. Cette étude est basée sur un principe simple: présenter un flux entaché d'erreurs (contrôlées et introduits volontairement) à des téléspectateurs, ceux-ci devant alors donner leurs appréciations sur la qualité visuelle et auditive du programme. L'étude a permis de dégager une première classification empirique des paramètres d'où l'apparition de ces nouveaux paramètres de QdS. Le projet RNRT EQUAST auquel nous avons participé regroupe 5 partenaires: TDF C2R, Thalès B&M , SODIELEC, LIEN (laboratoire d'électronique de l'UHP) et le LORIA). Son but est la réalisation d'un outil de mesure fondé sur l'intégralité de la nouvelle norme [9]. Du fait du nombre de traitements à réaliser, la conception puis la réalisation d'un tel outil sous la forme d'un système enfoui (System on Chip ou SoC) nécessite une analyse précise des besoins. En effet, malgré le lancement officiel de la TNT, aucun équipement du marché n'implante l'ensemble des paramètres d'évaluation de la qualité. Ceci est aussi bien dû aux coûts de réalisation importants qu'aux contraintes temps réel fortes issues de l'analyse « en direct » du flux.

1.2 Conception électronique de systèmes enfouis

La conception d'un système enfoui concerne des éléments matériels et logiciels et intègre des plusieurs types de contraintes sur l'architecture ou sur les aspects temps réel. La difficulté est renforcée par l'utilisation de la reconfiguration dynamique matérielle. Le concepteur doit partitionner l'application et déterminer la meilleure implantation de chaque partie de l'application en fonction du temps, du coût, etc... Ces éléments rendent la conception de systèmes enfouis une tâche très délicate et extrêmement complexe.

La chaîne standard de conception d'un système enfoui est constituée de plusieurs étapes. Dans un premier temps, le concepteur décrit globalement le système et teste les fonctionnalités de cette description. Le test des fonctionnalités se fait essentiellement par simulation (éventuellement par model-checking).

Cette première étape achevée, le concepteur détermine les parties matérielles et logicielles du système. Ces deux parties sont ré-écrites séparément (par des équipes séparées) dans un langage « haut-niveau » pour la partie logicielle (souvent C) et dans un langage permettant la synthèse de circuit pour la partie matérielle (souvent VHDL). Les nouvelles descriptions de l'application ne sont pas reliées avec la description fonctionnelle de celle-ci, rien ne garantit le respect du cahier des charges initial. Il faut donc à nouveau tester les deux parties du système, afin de vérifier le respect des fonctionnalités.

Enfin, la réunion des parties logicielle et matérielle provoquent souvent des ré-implantations pour cause de non-conformité entre les interfaces ou entre les délais de communication.

SystemC [11] se propose de simplifier cette conception en utilisant un seul langage pour la simulation à la synthèse de circuit. SystemC est une bibliothèque C++ permettant aussi bien la description de composants électroniques à bas niveau (niveau RTL) que la modélisation générale d'une architecture sans entrer dans les détails techniques de l'implantation. L'architecture peut être décrite dans un langage commun et testée de manière très simple. La solution SystemC n'est cependant pas entièrement satisfaisante [7,8] car il n'y a pas de lien formel entre le niveau abstrait et le niveau RTL: des erreurs peuvent donc apparaître lors du passage entre ces deux niveaux de description.

1.3 Contribution et résultats

Dans le cadre de ce projet, des parties critiques du système (certains paramètres particulièrement complexes) ont été développées en premier lieu. Ce choix a été fait à cause du nombre important de paramètres à développer et de la complexité de certains paramètres particuliers. Nous avons validé les implantations réalisées par des modèles B faisant le lien entre le cahier des charges et les architectures électroniques construites.

Nous avons proposé une méthode de conception fondée sur la preuve et reposant en partie sur la méthode B événementielle (également utilisée dans [4]), afin de valider l'ordonnancement des tâches réalisant les divers traitements. Pour cela, nous extrayons des propriétés invariantes sur les données traitées et en déduisons des informations sur les traitements (paramètres) associés. La preuve nous garantit la cohérence du système et la notion de raffinement nous permet sa construction incrémentale qui permet de produire une hiérarchie des paramètres.

Enfin, nous avons proposé une traduction de nos modèles B vers du code SystemC permettant la construction automatique des circuits tout en conservant les propriétés des modèles.

Par ailleurs, deux prototypes (AMETHYST II et SDVB-T-M) sont issus du projet EQUAST. Ces deux prototypes sont opérationnels, implantent la totalité de la norme d'analyse de qualité DVB et sont en passe de devenir des produits du catalogue des partenaires industriels du projet.

La suite du présent article présente plus en détails les premières études de cas menées, leurs motivations et le gain de ces premiers travaux. Nous présenterons ensuite notre approche globale du problème puis nous finirons en présentant notre traduction systématique de (certains) modèles B en du code SystemC ainsi que la technique de validation de cette traduction.

2 Acquisition des compétences des domaines de la TNT et de l'électronique

2.1 Communications entre les partenaires

La diversité des partenaires ne facilite pas la communication en début de projet: les domaines de compétence étaient éloignés (informatique, électronique, DVB) et des vocabulaires différents étaient employés pour décrire des concepts similaires. Par ailleurs, la norme DVB était très technique et sa lecture suscitait de nombreuses interrogations. Chaque partenaire avait par conséquent des difficultés à appréhender ce que pouvait apporter les autres et ce qu'ils voulaient.

Du fait de la complexité et du nombre de paramètres, les électroniciens se sont focalisés sur certains paramètres particulièrement complexes sans avoir de vision globale de l'application. Nous avons donc décidé de modéliser parallèlement ces premiers développements, afin de comprendre les méthodes de chacun sur des exemples communs et de comparer les résultats obtenus.

2.2 Étude de quelques paramètres

Une première expérience a consisté en la validation a posteriori d'une conception électronique

implantant un paramètre chargé de calculer le débit de transmission du flux (nombre d'octets par seconde par exemple). Cette étude de cas est présentée intégralement dans [6]. La conception de l'architecture et son implantation ont entièrement été réalisées par les électroniciens. Nous avons validé les choix faits en montrant que l'implantation est bien un raffinement du système décrit par le cahier des charges, une fois sa conception terminée.

Une autre étude de cas est l'implantation d'un décodeur virtuel DVB-T permettant de vérifier la bonne décodabilité du signal reçu. Ce point de la norme étant très complexe, l'implantation s'est basée sur nos modèles B qui expliquaient, grâce au raffinement, le fonctionnement du décodeur virtuel.

Un dernier exemple est l'acquisition de la synchronisation par un récepteur. Les récepteurs branchés sur le réseau reçoivent un flôt continu d'octets et, pour pouvoir décoder les données, ils doivent se repérer dans le flôt. Un octet spécifique (l'octet de synchronisation) débute chaque paquet du flôt et sert de repère. Après la réception de 5 octets de synchronisation successifs corrects, le récepteur est considéré comme synchronisé. L'acquisition de la synchronisation est la base de toute l'analyse de qualité et nous avons montré que, sous couvert d'un flôt correctement structuré, la synchronisation était toujours correctement détectée.

2.3 Établissement du protocole de modélisation

Ce premier travail a permis d'établir le dialogue entre les partenaires du projet et de se faire une idée des problèmes de conception électronique. Nous avons acquis une meilleure maîtrise du domaine et des problèmes techniques (efficacité, temps réel) et nous avons entrepris un développement complet du système en intégrant progressivement chaque paramètre et en maintenant un dialogue avec les autres partenaires afin de valider chaque décision <<système>>.

3 Modélisation incrémentale du système globale

Le développement complet de l'étude de cas représente 7 modèles B liés par la relation de raffinement. Comme annoncé précédemment, la complexité du système est distribuée au fur et à mesure du développement.

3.1 Principes généraux

Notre méthode de conception se base sur plusieurs principes:

- Le premier modèle abstrait est une simple « traduction » de la norme et est l'essence même de tout le développement. Cette traduction n'est pas complète et seuls les aspects fondamentaux du système sont représentés dans ce premier modèle, basé sur la compréhension globale de la norme après une première étude.
- Nous ajoutons, par raffinements successifs du modèle, les paramètres de la norme et les détails du système (environnement et outil). Ces paramètres sont ajoutés naturellement lorsque le modèle est suffisamment détaillé pour permettre la réalisation des traitements qui sont associés. En plus de permettre la cohérence d'un modèle à un autre, le raffinement permet une classification naturelle suivant le degré de compréhension et de précision du modèle.
- Nous utilisons des propriétés invariantes afin de prouver une dépendance (ou non) entre les paramètres ajoutés et les paramètres déjà présents dans le modèle. Les preuves nous garantissent la cohérence interne des modèles mais aussi la cohérence entre les divers raffinements.
- Nous proposons une interprétation du modèle sous forme d'arbre qui permet de débattre avec les spécialistes du domaine et de vérifier la bonne compréhension du cahier des charges. Cette

étape est très importante et amène à faire un certain nombre d'allers et retours entre le modèle et les documents normatifs pour corriger ou compléter le modèle abstrait.

3-2 Relations entre paramètres

Pour chaque paramètre X de la norme nous introduisons une variable VX dans le modèle qui représente l'état du paramètre associé:

- VX = OK signifie que le paramètre X a été évalué et qu'il est correct.
- VX = KO signifie que le paramètre X a été évalué et qu'une erreur a été détectée.
- VX = IND signifie que le paramètre X n'a pas encore été évalué.

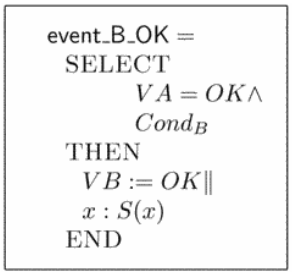
Le but de nos modèles est d'expliquer le système dans sa globalité, par raffinement. Nous nous attachons donc à décrire les relations existantes entre les différents paramètres de la norme et en particulier les relations permettant de définir un ordonnancement et de hiérarchiser l'ordre des traitements. Nous introduisons la notion de dépendance entre paramètres comme une relation invariante entre les variables modélisant ces paramètres.

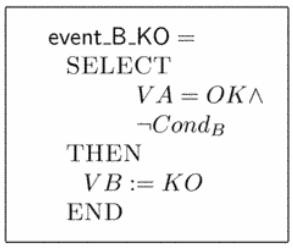
On dit qu'un paramètre B dépend d'un paramètre A lorsque:

$$B < A \equiv (VB \neq IND) \Rightarrow (VA = OK)$$

Cette propriété traduit le fait que le paramètre B ne peut être évalué ($VB \neq IND$) que si A a déjà été évalué et que le résultat de cette évaluation est correct ($VA = OK$). Il apparaît clairement que le paramètre B peut être indéterminé car l'évaluation de A n'est pas encore finie ou bien que le paramètre A signale une erreur et que, par conséquent, l'évaluation du paramètre B n'a pas de sens.

La dynamique du système est représentée par le déclenchement des événements. Les gardes des événements respectent les propriétés invariantes d'ordonnancement du modèle tout en modélisant l'évaluation des paramètres comme dans les exemples suivants:





Dans ces événements, le paramètre B ne peut être évalué qu'une fois le paramètre A correctement évalué. La valeur du paramètre B dépend alors de critères qui sont propres à la donnée traitée, ces critères étant représentés ici par la condition $Cond_B$. Enfin, lors de l'évaluation de B, le système met à jour les variables n'étant pas des paramètres (substitution x:S(x)). Bien entendu, la mise à jour de ces variables (mémoire, compteur, ...) n'a un intérêt que lorsque l'évaluation peut continuer c'est pourquoi la substitution x:S(x) n'est présente que dans l'événement event_B_OK.

L'ensemble des relations de dépendance constitue un graphe acyclique (s'il n'y a pas d'incohérences dans le cahier des charges) entre les paramètres. Le déclenchement des événements construit une valuation des paramètres respectant les propriétés invariantes déjà définies comme le présente la figure ci-dessous. Les paramètres situés au bas de ``l'arbre'' dépendent des paramètres constituant le chemin vers la racine (branche) et le paramètre-racine

ne dépend d'aucun paramètre.

L'arbre ainsi obtenu est un reflet de l'état du système, on peut suivre la progression de l'évaluation des paramètres du sommet vers les feuilles et connaître l'état de chaque paramètre au cours du temps. Certaines branches de l'arbre sont éventuellement bloquées par un paramètre incorrect alors que d'autres branches poursuivent leur évaluation jusqu'aux feuilles.

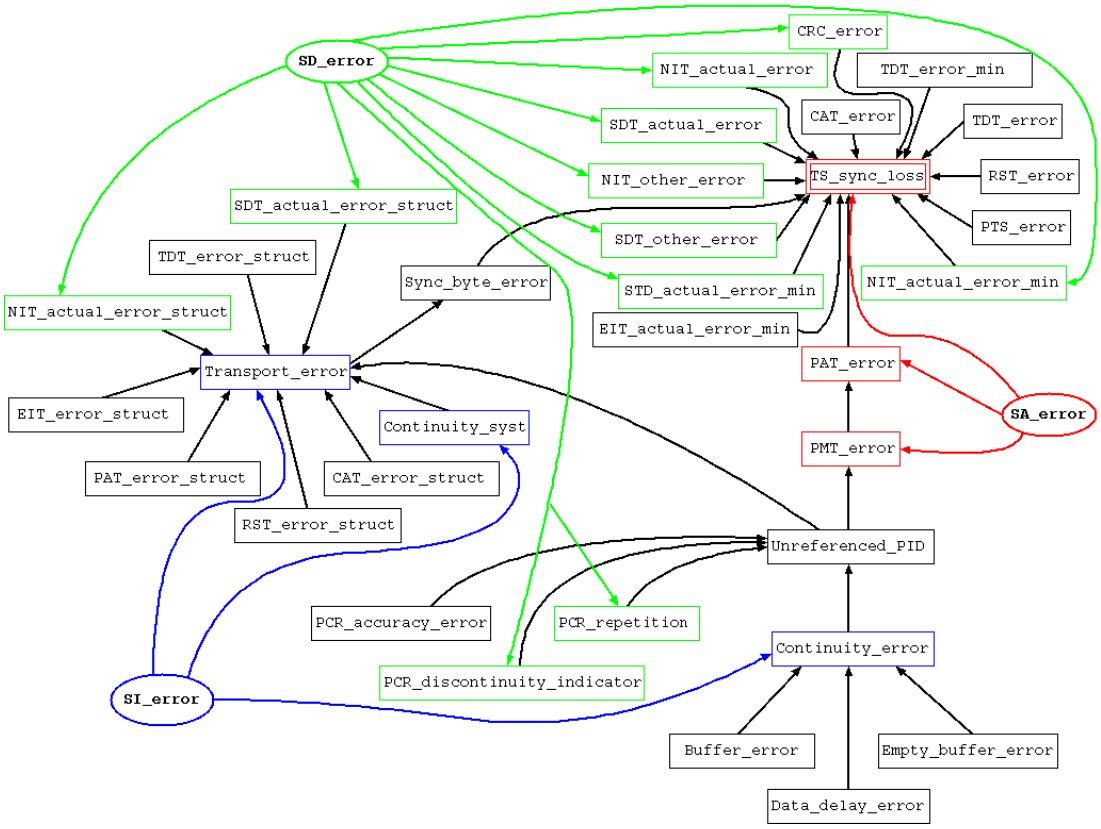
3.3 Modèle final et justification de la QdS

Le modèle final de notre développement formel contient l'intégralité des paramètres de la norme ainsi que les paramètres de synthèse (QdS). La figure suivante présente la hiérarchie finalement obtenue. La complexité de cette hiérarchie finale illustre bien l'intérêt du raffinement du point de vue de la distribution de complexité.

La figure présente également les trois paramètres de QdS: SA_error, SD_error et SI_error. Le paramètre SA_error est le paramètre chargé d'indiquer un dysfonctionnement grave de réception. Ce paramètre de synthèse et constitué de trois paramètres dont la racine de la hiérarchie formelle. Il est très clair que l'apparition d'erreurs à ce niveau a immédiatement de lourdes conséquences.

D'un autre côté, le paramètre SI_error indique de légères perturbations du signal. Ce paramètre est composé de « feuilles » et se situe donc assez bas dans la hiérarchie formelle. Il est clair que les erreurs détectées à ce niveau n'ont pas un impact trop important sur la réception, la majeure partie des traitements ayant déjà eu lieu.

Le dernier paramètre de synthèse, SD_error, doit permettre de suivre la dégradation du réseau passant de faiblement perturbé à très perturbé. Le paramètre en question est donc composé d'un grand nombre d'indicateurs répartis à tous ls niveaux de la hiérarchie et permettant un survol des principales causes de mauvais fonctionnement.



Les paramètres QdS ont été introduits à la suite de résultats empiriques pour permettre une synthèse plus rapide de l'état d'un réseau de diffusion. Notre interprétation, grâce aux positions des paramètres de base constituant ces paramètres QdS dans la hiérarchie, est concordante avec les résultats empiriques. Ce résultat est en soi intéressant mais a également permis de convaincre les industriels du projet de l'intérêt et de la validité de notre démarche. Les détails de ce résultat sont présentés dans [2].

4 Traduction de modèles B en programmes SystemC

La traduction de nos modèles en design SystemC est automatique et repose sur des principes simples qui permettent un code lisible et performant.

4.1 Principes généraux

La traduction d'un modèle se fait de manière très simple. Le modèle en question définit une hiérarchie de tâches sous la forme d'une relation. Chaque noeud de la hiérarchie est traduit en un processus SystemC. La relation de dépendance est implantée à l'aide de canaux de communication (ou channels). Chaque processus implantant une tâche a deux canaux booléens de sortie utilisés pour implanter les valeurs OK et KO des modèles B, la valeur IND étant implantée par l'absence de signal sur les deux canaux. Pour implanter la dépendance, on dote les processus SystemC d'autant de canaux d'entrée booléens que de parents directs. L'écriture par les processus pères de ces canaux provoque le réveil du processus fils qui réalise ses traitements s'il a reçu un acquittement de l'ensemble de ses parents.

La figure ci-dessous présente un exemple simplifié d'architecture obtenue par traduction d'un modèle B de hiérarchie. Chaque processus engendré est muni de deux canaux de sortie qui sont reliés à ses éventuels descendants. Réciproquement, chaque processus est relié à l'ensemble de ses parents et ne réalise ces traitements qu'une fois l'ensemble des acquittements de ses parents reçus. Les détails de la traduction ont été présentés dans [3] (de légères modifications ont été apportées depuis) et ne peuvent être entièrement repris ici faute de place.

4.2 Cohérence de la traduction

Nous proposons de prouver la cohérence de notre traduction par raffinement. Ce travail, en cours, nous permettra de montrer que la fonction de traduction revient à une série de raffinements. Pour vérifier cette idée nous avons modélisé l'ordonnanceur SystemC. Cette modélisation nous permet d'avoir un modèle générique de la simulation d'un programme SystemC. Ce modèle peut alors être instancié pour un design particulier. Nous pouvons alors raisonner sur les programmes SystemC car nous définissons leurs sémantiques à l'aide de notre modèle.

Le modèle source de la traduction est un modèle B (Hiérarchie) représentant une hiérarchie de tâches. En appliquant notre fonction de traduction à ce modèle, on obtient un programme SystemC (P). Le modèle B du scheduler SystemC (Scheduler) est instancié avec le programme P pour produire un modèle B (ExecutionP) représentant l'exécution par le scheduler du programme P traduit.

Par ailleurs, le modèle B Hiérarchie est raffiné et les principales constructions SystemC sont introduites. On obtient finalement la description B Architecture raffinant une hiérarchie de tâches et décrivant l'implantation SystemC réalisée.

Si nous montrons que le modèle B ExecutionP, instancié, a les mêmes comportements que le modèle Architecture, nous aurons prouvé la validité de notre traduction. La figure suivante présente l'intégralité de notre démarche, sans pour autant donner tous les détails techniques de réalisation.

4.3 Gain de la traduction automatique

La traduction automatique permet de régler plusieurs problèmes. Tout d'abord, la traduction engendre du code SystemC RTL qui est synthétisable. De ce fait, les problèmes de réécriture dûs aux problèmes de synthèse de circuit sont résolus. Par ailleurs, le lien entre le cahier des charges et l'architecture est assuré par les raffinements ayant permis de construire la hiérarchie de tâches. La traduction étant validée, le code SystemC est une implantation correcte des modèles et donc du cahier des charges. Enfin, le problème de composition de modules développés indépendamment est contourné car les éventuels modules développés séparément se greffent dans le code SystemC produit qui respecte l'ordonnancement établi dans les modèles.

5 Conclusions et perspectives

Nos travaux au sein du projet Equast nous ont permis de confronter la méthode B événementielle à une étude de cas original qui n'aboutit pas sur la production de code informatique mais sur la génération de circuits électroniques. Les différentes modélisations conduites au sein de ce projet ont permis de définir une méthode de conception fondée sur la preuve et assurant une continuité entre cahier des charges et architecture. La validation de la traduction de modèles B en du code SystemC par raffinement est également un point intéressant qui offre une nouvelle approche. Nous sommes en possession d'un modèle générique d'exécution de SystemC qui permet de raisonner sur ces programmes et une des pistes intéressantes à poursuivre est celle de la comparaison de deux programmes SystemC.

6 Références

[1] J.-R. Abrial. The B-Book - Assigning Programs to Meanings. Cambridge University Press, 1996

[2] D. Abraham, D. Cansell, P. Ditsch, D. Méry et C. Proch. The Challenge of QoS for digital television services. EBU Technical Review, 302, Avr 2005.

[3] D. Cansell, J.-F. Culat, D. Méry et C. Proch. Derivation of SystemC code from abstract system models. In Forum on specification & Design Languages – FDL'04, Lille, France, Sep 2004.

[4] D. Cansell, S. Hallerstede et Y. Zimmermann. Construction sûre de systèmes électoniques, Génie Logiciel, Juin 2004, 69, pages 38-44.

[5] D. Cansell et D. Méry. Logical foundations of the B method. In Computers and Informatics n°22, 2003.

[6] D. Cansell, C. Tanougast, Y. Berviller, D. Méry, C. Proch, H. Rabah et S. Weber. Proof-based design of a microelectronic architecture for MPEG-2 bit-rate measurement. In Forum on specification & Design Languages – FDL'03, Franckfurt, Germany, Sep 2003.

[7] A. Gawanmeh, A. Habibi et S. Tahar. Enabling SystemC Verification using Abstract State Machines. In Languages for Formal Specification and Verification, Forum on Specification &

Design Languages – FDL'04, Lille, France, Sep 2004.

[8] A. Salem, Formal Semantics of Synchronous SystemC. In Design Automation and Test in Europe – DATE'03 , Munich, Germany, Mar 2003.

[9] Digital Video Broadcasting DVB; Measurement guidelines for DVB systems. ETSI TR 101 290 v1.2.1. http://www.etsi.org, Avr 2001.

[10] Digital Video Broadcasting DVB; Implementation guidelines for the use of MPEG-2 Systems, Video and Audio in satellite, cable or terrestrial broadcasting applications. ETSI TR 101 154. http://www.etsi.org, 2002.

[11] SystemC 2.0.1; Official web site of SystemC Community. http://www.systemc.org/, 1999.