POLYPUBLIE

Validation matérielle d'une architecture générique de réseaux avioniques basée sur une gestion modulaire de la redondance

José-Philippe Tremblay

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

Date	2016
Type	Mémoire ou thèse / Dissertation or Thesis Thèse de doctorat/Ph.D. thesis
URL de PolyPublie PolyPublie URL	https://publications.polymtl.ca/2143/
Conditions d'utilisation Terms of use	© Tous droits réservés / All rights reserved
Référence Citation	Tremblay, J.-P. (2016). Validation matérielle d'une architecture générique de réseaux avioniques basée sur une gestion modulaire de la redondance [Thèse de doctorat, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/2143/

Détails de la thèse
Thesis details

Directeurs de recherche Advisors	Yvon Savaria, & Claude Thibeault
Programme Program	génie électrique

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal
https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

VALIDATION MATÉRIELLE D'UNE ARCHITECTURE GÉNÉRIQUE DE RÉSEAUX AVIONIQUES BASÉE SUR UNE GESTION MODULAIRE DE LA REDONDANCE

JOSÉ-PHILIPPE TREMBLAY

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION

DU DIPLÔME DE PHILOSOPHIAE DOCTOR

(GÉNIE ÉLECTRIQUE)

AVRIL 2016

© José-Philippe Tremblay, 2016.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

VALIDATION MATÉRIELLE D'UNE ARCHITECTURE GÉNÉRIQUE DE RÉSEAUX AVIONIQUES BASÉE SUR UNE GESTION MODULAIRE DE LA REDONDANCE

présentée par : TREMBLAY José-Philippe

en vue de l'obtention du diplôme de : Philosophiae Doctor

a été dûment accepté par le jury d'examen constitué de :

M. SAWAN Mohamad, Ph. D., président

M. SAVARIA Yvon, Ph. D., membre et directeur de recherche

M. THIBEAULT Claude, Ph. D., membre et codirecteur de recherche

M. BELTRAME Giovanni, Ph. D., membre

M. AUDET Daniel, Ph. D., membre externe

DÉDICACE

À Claudia, Chloé et Camille avec tout mon amour ...

REMERCIEMENTS

Je te tiens d‟abord à remercier M. Savaria et M. Thibeault, respectivement directeur et co-directeur de doctorat, pour leur disponibilité, leur support et divers conseils menant à terme avec succès mes travaux de recherche. Je tiens également à souligner le soutien financier de Bombardier, principal partenaire du projet AVIO402. Je remercie également mes coéquipiers du GRM et partenaires du projet AVIO402 pour leur support et encadrement lors de la réalisation du projet. Je souligne plus particulièrement l‟appui de Safwen Bouanen et de Davide Trentin pour leur précieuse collaboration. Je termine par remercier ma famille et mes amis qui m‟ont soutenu tout au long de mes études.

RÉSUMÉ

Les systèmes avioniques ne cessent d'évoluer depuis l'apparition des technologies numériques au tournant des années 60. Après le passage par plusieurs paradigmes de développement, ces systèmes suivent maintenant l'approche « Integrated Modular Avionics » (IMA) depuis le début des années 2000. Contrairement aux méthodes antérieures, cette approche est basée sur une conception modulaire, un partage de ressources génériques entre plusieurs systèmes et l'utilisation plus poussée de bus multiplexés. La plupart des concepts utilisés par l'architecture IMA, bien que déjà connus dans le domaine de l'informatique distribuée, constituent un changement marqué par rapport aux modèles antérieurs dans le monde avionique. Ceux-ci viennent s'ajouter aux contraintes importantes de l'avionique classique telles que le déterminisme, le temps réel, la certification et les cibles élevées de fiabilité. L'adoption de l'approche IMA a déclenché une révision de plusieurs aspects de la conception, de la certification et de l'implémentation d'un système IMA afin d'en tirer profit. Cette révision, ralentie par les contraintes avioniques, est toujours en cours, et offre encore l'opportunité de développement de nouveaux outils, méthodes et modèles à tous les niveaux du processus d'implémentation d'un système IMA. Dans un contexte de proposition et de validation d'une nouvelle architecture IMA pour un réseau générique de capteurs à bord d'un avion, nous avons identifié quelques aspects des différentes approches traditionnelles pour la réalisation de ce type d'architecture pouvant être améliorés. Afin de remédier à certaines des différentes lacunes identifiées, nous avons proposé une approche de validation basée sur une plateforme matérielle reconfigurable ainsi qu'une nouvelle approche de gestion de la redondance pour l'atteinte des cibles de fiabilité. Contrairement aux outils statiques plus limités satisfaisant les besoins pour la conception d'une architecture fédérée, notre approche de validation est spécifiquement développée de manière à faciliter la conception d'une architecture IMA.

Dans le cadre de cette thèse, trois axes principaux de contributions originales se sont dégagés des travaux exécutés suivant les différents objectifs de recherche énoncés précédemment. Le premier axe se situe au niveau de la proposition d'une architecture hiérarchique de réseau de capteurs s'appuyant sur le modèle de base de la norme IEEE 1451. Cette norme facilite l'intégration de capteurs et actuateurs intelligents à tout système de commande par des interfaces normalisées et génériques. Afin d'être applicable dans le domaine avionique pour le traitement d'applications

critiques, nous avons introduit dans notre architecture de réseau de capteurs génériques notre propre gestion de la redondance permettant entre autres d'atteindre des niveaux plus élevés de fiabilité ainsi que de modifier le niveau de fiabilité selon les exigences. De manière à respecter les différentes contraintes liées au développement de systèmes avioniques, l'architecture proposée permet la personnalisation de la bande passante offerte, du niveau de fiabilité recherché, de la quantité de ressources consommées et du type de capteurs ou actuateurs utilisés. La modélisation de plusieurs configurations est présentée dans la thèse afin de pouvoir comparer les différentes architectures et en faire ressortir les avantages et inconvénients. Au niveau de la fiabilité, nos analyses montrent la polyvalence de notre architecture en permettant la modulation de la fiabilité totale du réseau grâce à la variation intégrée du nombre de composants et de bus, contrairement aux approches traditionnelles qui ne possèdent aucun mécanisme intrinsèque pour augmenter la fiabilité du réseau rendant leur application inefficace dans un contexte d'applications critiques. Au niveau de la consommation de ressources, notre architecture permet des gains raisonnables face aux architectures classiques.

Le second axe porte sur la proposition d'une plateforme matérielle de design et de validation de systèmes avioniques spécifiquement adaptée aux besoins de l'approche IMA. La constante migration vers les prochaines générations de systèmes avioniques requiert toujours l'adoption de nouvelles technologies de communication de données, comme ce fut le cas pour les protocoles ARINC 825 et ARINC 664 introduits il y a déjà quelques années. Les caractéristiques et performances peuvent varier énormément d'une norme à l'autre. L'intégration de ces protocoles peut donc demander un effort considérable afin de respecter les différentes contraintes du domaine avionique. Bien que des composants commerciaux soient disponibles, l'utilisation d'une implémentation matérielle personnalisée et générique peut s'avérer extrêmement utile dans la validation d'architectures ou d'analyses théoriques et l'identification des cas problématiques. À travers le cas de figure étudié, nous montrons les différents avantages de l'utilisation d'une plateforme matérielle, bien que certaines analyses du processus de design et de validation requièrent toujours l'emploi de modélisation logicielle ou mathématique. Le caractère réutilisable de la plateforme FPGA ainsi que l'architecture générique sélectionnée permet ultimement de réduire significativement l'effort de design et de validation de ce type de système. La possibilité offerte de validation en temps réel peut également s'avérer particulièrement utile dans le domaine. Dans le cas où une approche propriétaire est privilégiée comme dans le cadre de notre

projet, une compréhension plus complète des différentes normes peut permettre de détecter des modes d'erreur difficile à cerner à l'aide de modèles mathématiques ou de tester de nouvelles fonctionnalités tel le regroupement de trames dans un contrôleur ARINC 825. L'exemple final du système de monitorage montre par contre que l'approche matérielle peut être moins adaptée pour faire face à certaines problématiques ou analyses requises motivant toujours l'emploi de la modélisation.

Au niveau de l'atteinte des cibles de fiabilité, les systèmes avioniques ont toujours majoritairement recours à la redondance pour en assurer le respect. Le troisième et dernier axe de contribution de cette thèse repose sur la présentation de notre approche de gestion de la redondance spécifique au domaine avionique permettant entre autres la sélection du niveau approprié de redondance de manière spécifique pour chacune des couches traversées. Notre approche est illustrée à travers le design et la validation d'un réseau de capteurs basé sur un réseau AFDX principal, supporté par plusieurs bus secondaires suivant la norme ARINC 825. Afin de capturer fidèlement les contraintes du domaine avionique, les points communs de défaillance ou couplages ont été inclus dans la modélisation de notre architecture et dans celle des approches de gestion de redondance classiques. Une première analyse vient démonter un gain de fiabilité substantiel face à l'approche de type « composant » ainsi qu'une fiabilité similaire à l'approche de type « système ». Une deuxième phase vient par contre démontrer la meilleure efficacité au niveau des ressources face à l'approche « système ». À l'aide de diverses analyses, la sensibilité du réseau face à un composant ou aux points de couplage vient faciliter la phase de design en permettant la sélection d'une configuration plus efficace par rapport aux contraintes spécifiées. L'architecture proposée prend le meilleur de chacune des méthodes classiques et élimine le plus de points communs de défaillance possible.

ABSTRACT

Since the emergence of digital technologies in the 1960s, avionics has evolved very rapidly. To reflect the technological evolution, several design approaches have been followed up to the Integrated Modular Avionics (IMA) introduced in the late 1990s. Unlike previous approaches, IMA is based on modular conception, generic resources sharing between several previously independent systems and further use of multiplexed busses. While the major aspects used in the IMA architecture is already known in other fields such as distributed computing, they represent a fundamental change in regards to traditional approaches for avionics. These new concepts are added to the existing constraints of classic avionic such as determinism, real time application, certification and stringent reliability targets. Several aspects of the design, certification and implementation of an IMA system must be revised in order to maximize its benefice. The adoption of the IMA thus requires the development of new approaches, models and tools at each level of the implementation process of an IMA system. In a context of proposition and validation of a new IMA architecture for a generic sensor network, we have identified several key aspects of the traditional approaches that could be enhanced for a better compatibility with the IMA approach. In order to overcome these identified deficiencies, we have proposed a global validation approach based on a reconfigurable material platform combined with a new approach to redundancy management required to reach the reliability targets. Unlike the static tools devised for the previous federated design approach, our approach was specifically devised to ease the design process under the IMA architecture.

Within the context of this thesis, three primary areas of original contributions have been identified from our work done within the objectives of project AVIO 402. The first contribution resides in the proposition of a hierarchical architecture for a sensor network base on the reference model of the IEEE 1451 standard. This standard facilitates the integration and control of any type of intelligent transducer to any control system through normalized generic interfaces. In order to be compliant with the avionics constraints for critical applications, we have introduced within our architecture our own redundancy management scheme allowing the required levels of reliability as well as the possibility to tailor the system‟s reliability depending on the requirements. To reach these different requirements, the architecture can be personalized in terms of available bandwidth, reliability, quantity of consumed resources and of the type of transducers used. The

modeling of several configurations is presented in this thesis in order to compare the different architectures and highlight the benefits and limitations of each. Our reliability analyses show the versatility of our architecture by allowing the reliability modulation of the network by varying the number of components and/or busses unlike previous implementation of the IEEE 1451 standard. The IEEE 1451 standard would be applicable to the avionics field without this last feature. In terms of resource consumption, our architecture offers significant gains compared to traditional approaches.

The second contribution lies in the proposition of a hardware platform specifically devised for the design and validation of avionics system following the IMA approach. The steady evolution towards next generation avionics systems requires the adoption of new communications technologies, such as the ARINC 825 and AFDX protocols introduced several years ago and currently widely used. The characteristics and performances of these emerging technologies can greatly vary from one another and their integration can necessitate a considerable effort while complying with avionics constraints. Even if commercials components are available, a custom material generic implementation can be extremely useful for the validation of architectures or theoretical analyses and to identify problematic cases. Through a case study, we show the different advantages and limitations of the use of our hardware platform. As a first benefit, the reusable nature of the platform in combination with the selected generic architecture can greatly reduce the design and validation effort for this type of system. The direct inclusion of the certification process can also provide the same benefit by using final implementation for testing and validation. The possibility of real time validation is also extremely useful in the avionics domain. In the case where a proprietary approach is preferred, a better comprehension of the different standards can help detect error source difficult to identify with mathematic model or to test new features such as our testing of frame packing strategies directly included within our ARINC 825 controller. Our final example based on our innovative monitoring system shows that the hardware platform is not always the best suited for certain types of analysis. We then conclude that even though software and mathematical analysis are still needed, a mixed approach would still bring a lot of benefits to the design and validation of avionics systems.

In order to reach the high reliability targets, avionics systems have mostly relied on redundancy. The third and last contribution presented in this thesis resides in the presentation of our redundancy management approach. Globally, our approach allows the individual customization

of each stage of the network and is demonstrated through the validation of a sensor network devised within the AVIO 402 project. In order to correctly capture the constraints of the avionics field, single points of failures or coupling have been introduced into the model of our architecture as well as traditional approach to redundancy management. A first analysis shows a substantial gain in reliability compared to the typical component approach and a similar level of reliability than the system redundancy management scheme. In a second step, a better efficiency of consumed resources is measured in respect to the system architecture. The analyses also enable the selection of a more efficient configuration by underlining the sensitivity of the network to those single points of failures. The proposed redundancy management scheme takes the best advantages from each classical approach and eliminates as many single points of failures as possible.

TABLE DES MATIÈRES

DÉDICACE iii

REMERCIEMENTS iv

RÉSUMÉ v

ABSTRACT viii

TABLE DES MATIÈRES xi

LISTE DES TABLEAUX xiv

LISTE DES FIGURES xv

LISTE DES SIGLES ET ABRÉVIATIONS xvii

CHAPITRE 1 INTRODUCTION 1

1.1 Mise en contexte 1

1.2 Problématique 5

1.3 Cadre de projet 6

1.4 Objectifs de recherche et contributions 9

1.5 Organisation de la thèse 11

CHAPITRE 2 RÉSEAUX AVIONIQUES DE CAPTEURS 13

2.1 IEEE 1451 14

2.1.1 Principes de base 14

2.1.2 Applications 16

2.2 Design et validation de systèmes IMA 20

2.2.1 Approche classique 20

2.2.2 Approches originales 23

2.2.3 Monitorage 27

2.3 ARINC 825 29

2.3.1 Principes de base 29
2.3.2 Usages 30
2.3.3 Regroupement de trames 31
2.4 Gestion de la redondance 34
2.4.1 Approche classique 34
2.4.2 Approche originale 36
2.4.3 Approches alternatives 36
CHAPITRE 3 ARTICLE 1 : A SYSTEM ARCHITECTURE FOR SMART SENSORS INTEGRATION IN AVIONICS APPLICATIONS 39
3.1 Introduction 40
3.2 Related Works 41
3.3 System Architecture 44
3.4 Reliability Evaluation 46
3.5 Results 47
3.6 Conclusion 51
CHAPITRE 4 ARTICLE 2 : HIERARCHICAL REDUNDANCY MANAGEMENT FOR AVIONIC NETWORKS 52
4.1 Abstract 52
4.2 Introduction 52
4.3 Avionics Networks Redundancy 54
4.3.1 Redundancy Management 54
4.3.2 Federated and Integrated Modular Architectures 57
4.4 Redundancy Management 59
4.5 Hierarchical Architecture Analysis 67
4.5.1 Reliability estimations 67

4.5.2 Complexity analysis 70
4.5.3 Coupling‟s impact on reliability 72
4.6 Conclusion 77
CHAPITRE 5 DISCUSSION GÉNÉRALE 79
5.1 Architecture générique de réseau avionique 79
5.1.1 Sommaire 79
5.1.2 Discussion 80
5.2 Validation de l‟approche mixte pour l‟intégration de réseaux avioniques 81
5.2.1 Sommaire 81
5.2.2 Méthodologie 82
5.2.3 Implémentation 86
5.2.4 Discussion 95
5.3 Gestion modulaire de la redondance 100
5.3.1 Sommaire 100
5.3.2 Discussion 101
CHAPITRE 6 CONCLUSION 103
6.1 Synthèse 103
6.2 Limitations et travaux futurs 105
RÉFÉRENCES 107

LISTE DES TABLEAUX

Tableau 2.1: Domaines d'application de la norme IEEE 1451 19

Tableau 2.2: Niveau de redondance et cibles de fiabilité [1] 34

Tableau 3.1: Modules complexity and failure rate 48

Tableau 4.1: Architecture's failure rates 69

Tableau 4.2: Number of components for the traditional architectures 70

Tableau 4.3: Number of components for the hierarchical architecture 71

Tableau 5.1: Requis 85

Tableau 5.2: Notation adoptée pour décrire les composants 88

Tableau 5.3: Complexité des composants 91

Tableau 5.4: Latence moyenne de trames 94

Tableau 5.5: Validation de réseau 95

Tableau 5.6: Trafic sous la stratégie de regroupement de trames *M-to-1* 99

LISTE DES FIGURES

Figure 1-1: Évolution des architectures avioniques [1] 1

Figure 1-2: Architectures distribuées A) analogique B) numérique 2

Figure 1-3: Exemple d"architecture fédérée 3

Figure 1-4: Exemple d"architecture IMA [1] 4

Figure 1-5: Architecture proposée [6] 8

Figure 2-1: Modèle de référence du IEEE 1451 [7] 15

Figure 2-2: Architecture réseau [12] 17

Figure 2-3: Architecture réseau [14] 18

Figure 2-4: Relation entre les principales normes avioniques [1] 22

Figure 2-5: Stratégie *M-to-1* 33

Figure 2-6: Architecture Com-Mon 35

Figure 3-1: IEEE 1451 Reference Model, adapted from [7] 42

Figure 3-2: IEEE 1451 Distributed Architecture 43

Figure 3-3: Custom global architecture 45

Figure 3-4: Custom TIM"s architecture 45

Figure 3-5: Custom NCAP"s architecture 46

Figure 3-6: Network Complexity Vs. Number of Transducers 49

Figure 3-7: Network Complexity Vs. Number of Transducers 50

Figure 3-8: Network Complexity Vs Number of Transducers 51

Figure 4-1: Federated Architecture 57

Figure 4-2: Airbus A320 flight deck display system [1] 58

Figure 4-3: IMA Architecture 59

Figure 4-4: A) Functional architecture, non-redundant version B) Associated reliability model 60

Figure 4-5: A) System redundant architecture, B) Associated reliability model 62

Figure 4-6: A) Component redundant architecture B) Associated reliability model 63

Figure 4-7: A) Hierarchically redundant architecture, B) Associated reliability model 64

Figure 4-8: TIM Structure A) Component Architecture B) Hierarchical Architecture 65

Figure 4-9: AFDX End System Specification 66

Figure 4-10: Architecture‟s unreliability as a function of the AFDX‟s unreliability for error detection 73

Figure 4-11: Architecture‟s unreliability as a function of the AFDX‟s unreliability for data correction 74

Figure 4-12: Architecture‟s unreliability as a function the coupling‟s unreliability for error detection 75

Figure 4-13: Architecture‟s unreliability against the coupling‟s unreliability for data correction 76

Figure 5-1: Méthodologie proposée 83

Figure 5-2: Arbre de panne du réseau complet 88

Figure 5-3: Arbre de panne d‟un TIM 89

Figure 5-4: Arbre de panne d‟un NCAP 89

Figure 5-5: Prototype matériel 92

Figure 5-6: Exemple de réseau 94

Figure 5-7: Composition de la trame 98

LISTE DES SIGLES ET ABRÉVIATIONS

ARP	Avionics Recommended Practices
CAN	Controller Area Network
DASC	Digital Avionics System Conference
FCC	Flight Control Computer
FMEA	Failures Modes and Effects Analysis
FPGA	Field Programmable Gate Array
FTA	Fault Tree Analysis
IMA	Integrated Modular Avionics
LRU	Line Replaceable Unit
LUT	Look Up Table
NCAP	Network Capable Application Processor
SEU	Single Event Upset
TEDS	Transducer Electronic Datasheet
TIM	Transducer Interface Module
TRL	Technological Readiness Level

CHAPITRE 1 INTRODUCTION

1.1 Mise en contexte

Dans le domaine aéronautique, l'application des technologies numériques a été introduite rapidement suite à l'amélioration marquée des performances et de la complexité des différents aéronefs civils. La Figure 1-1 représente l'évolution des architectures avioniques depuis l'introduction des composants électroniques au début des années 60. Auparavant, la quasi-totalité des systèmes aéronautiques dans l'industrie civile était implémentée de manière électromécanique. Plusieurs des appareils développés à l'aide des approches distribuées analogique et numérique sont toujours en service, tels le Boeing 707 et les premiers modèles de Boeing 737, bien que leur nombre diminue rapidement en raison du coût de maintenance exorbitant de ces systèmes vieillissants [1].

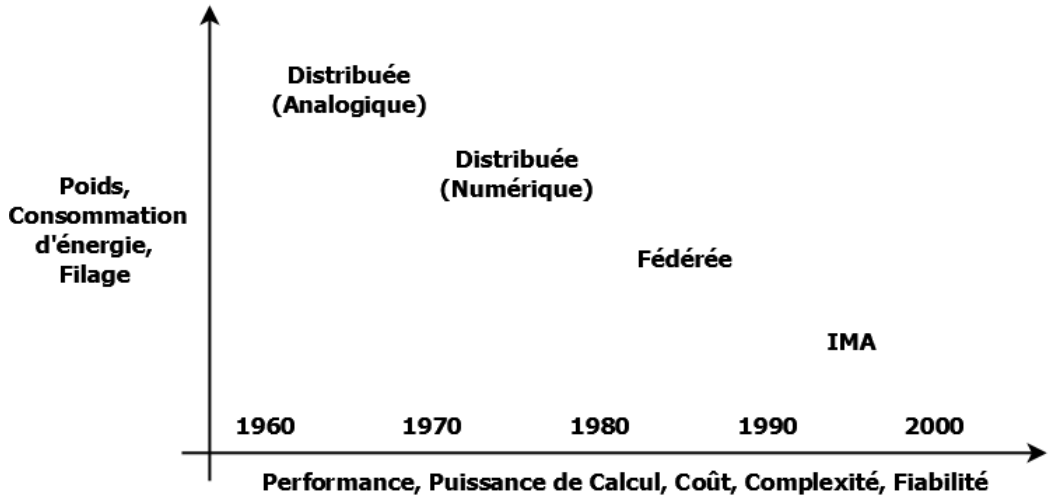


Figure 1-1: Évolution des architectures avioniques [1]

Les deux premiers paradigmes de développement de systèmes avioniques, basés sur l'isolation complète de chaque sous-système, n'ont comme distinction que la technologie utilisée pour leur implémentation. L'évolution technologique des années 70, incluant l'apparition de bus de données numériques, permit entre autres d'augmenter la performance de ces systèmes en maintenant la même approche de développement d'architecture. À ce moment, les connexions analogiques directes ont été remplacées par des bus de données unidirectionnels, tel le ARINC 429 [2], le protocole le plus populaire à l'époque. Malgré un faible débit de 100 kbps, cette

norme est toujours importante pour l'interfaçage de composants plus simples et comme manière d'introduire de la dissimilitude dans les systèmes critiques. Malgré l'introduction de bus pour l'architecture numérique, le raccordement des instruments à ce stade ne pouvait toujours se faire que de manière analogique. La grande popularité de l'ARINC 429 a mené à un premier effort de standardisation des équipements permettant éventuellement une évolution vers l'approche fédérée. Bien que l'augmentation des performances des transferts numériques à l'intérieur d'un même système ait mené à une utilisation grandissante de processeurs numériques dans de nouveaux systèmes, le développement de chaque système est demeuré orienté vers la réalisation d'une seule fonction indépendante. La figure suivante fournit plus de détails sur un exemple d'architecture distribuée réalisée selon les approches analogiques et numériques. À la Figure 1-2, la distinction majeure entre les deux modèles repose sur le remplacement de nombreux fils indépendants par des bus unidirectionnels de données numériques.

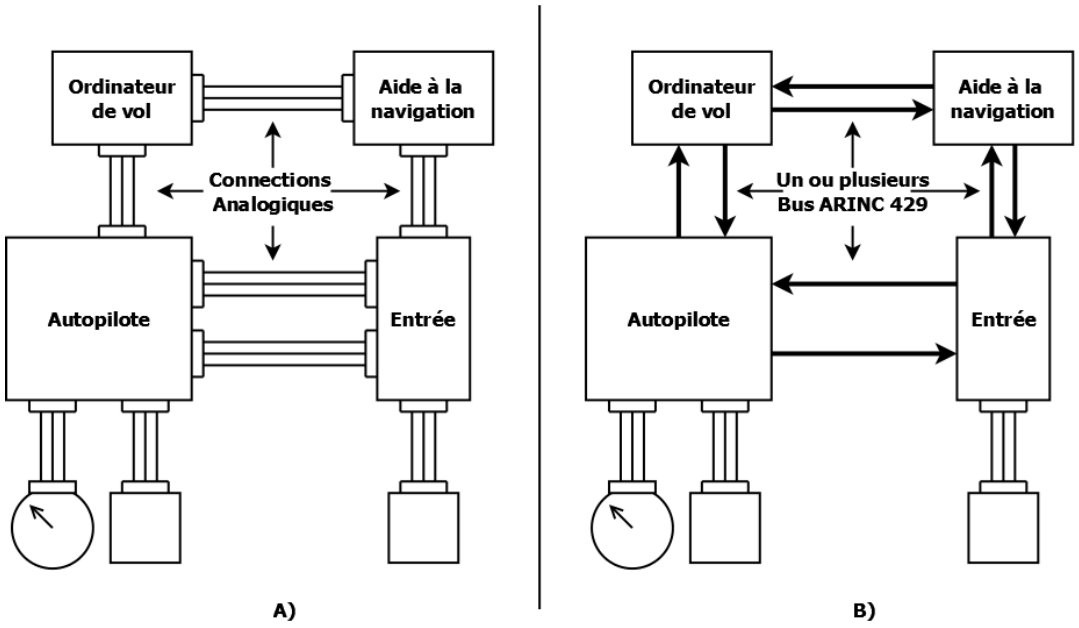


Figure 1-2: Architectures distribuées A) analogique B) numérique

L'approche de conception introduite après l'approche distribuée, communément appelée l'architecture fédérée, est basée sur la séparation des fonctions et du matériel et permet un processus simple de validation et d'intégration avec de simples assurances de respect des contraintes de bande passante et de synchronisation. Avec l'approche fédérée, les systèmes avioniques ont graduellement été intégrés et groupés selon les différents domaines d'application (commande de vol, carburant, train d'atterrissage …). La Figure 1-3 montre un exemple

d‟architecture fédérée typique tel que l‟on peut retrouver à bord du Boeing 777. Cette architecture a été adoptée massivement par l‟industrie à partir des années 80.

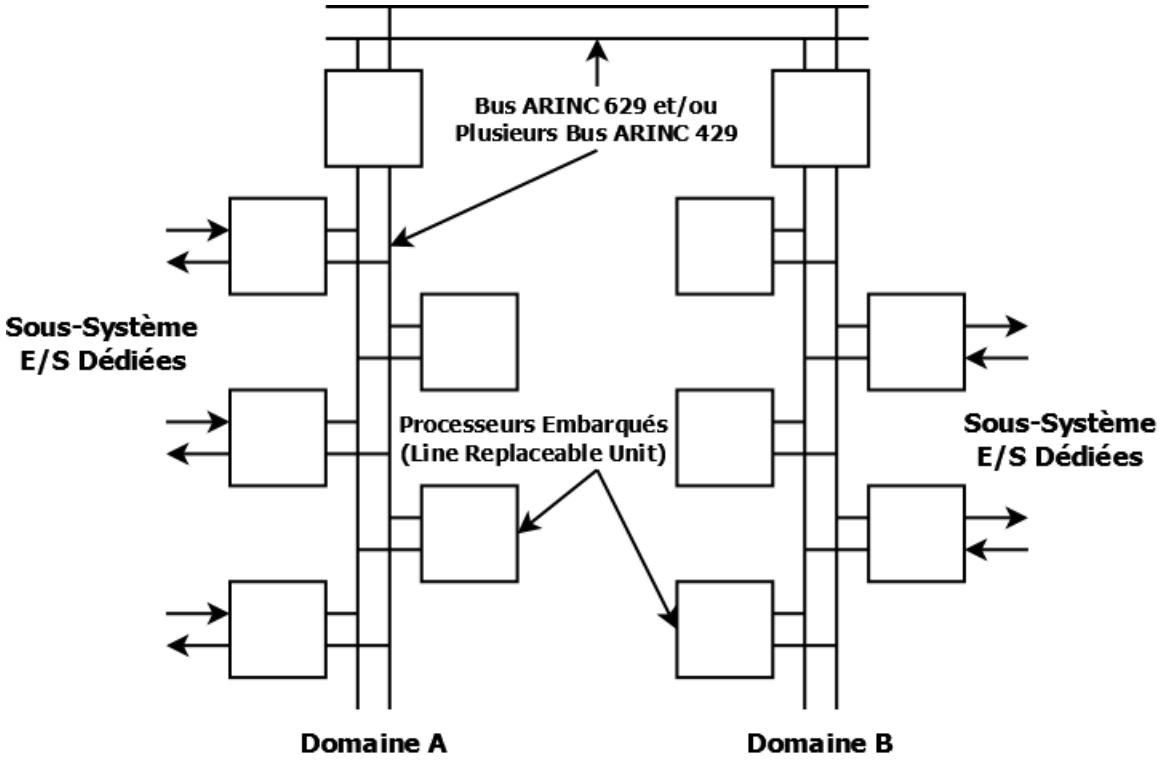


Figure 1-3: Exemple d‟architecture fédérée

L‟approche fédérée traditionnelle repose donc sur le développement de composants fonctionnels indépendants connectés par des bus unidirectionnels et par d‟autres signaux de contrôle dédiés. À ce stade de développement, l‟équipement avionique se décline principalement sous la forme de designs propriétaires offrant une solution dédiée à une fonctionnalité précise limitant l‟utilisation de ces composants dans d‟autres champs d‟activité. Les limitations de cette architecture ont commencé à apparaître avec l‟avènement des nouvelles technologies de communication et la multiplication des composants de toutes sortes. En effet, les avancées technologiques en électronique permettant une intégration toujours plus efficace des différents composants ont permis de grandement augmenter les différentes fonctionnalités requises à bord d‟un aéronef. L‟utilisation croissante de ces technologies numériques a donc entraîné une augmentation marquée de la complexité des systèmes. En suivant l‟approche fédérée, ces nouvelles fonctionnalités ont entraîné une augmentation significative du nombre d‟équipements de toutes sortes (Bus, CPU, LRU, FCC). Cette complexité croissante des systèmes avioniques a entrainé

une augmentation substantielle des coûts de développement et d'exploitation. La réduction et la maitrise de cette complexité permettent donc un meilleur contrôle des coûts de réalisation, de maintenance et de mise à niveau. Pour faire face à ces nouveaux défis, un nouveau paradigme de développement a donc été nécessaire. À l'aide des nouvelles technologies aux capacités toujours grandissantes, il est maintenant possible de pallier aux limites de l'architecture fédérée à l'aide d'une nouvelle architecture intégrée communément appelée l'architecture intégrée modulaire ou « Integrated Modular Avionics » (IMA). Un exemple typique de l'architecture IMA est présenté à la Figure 1-4.

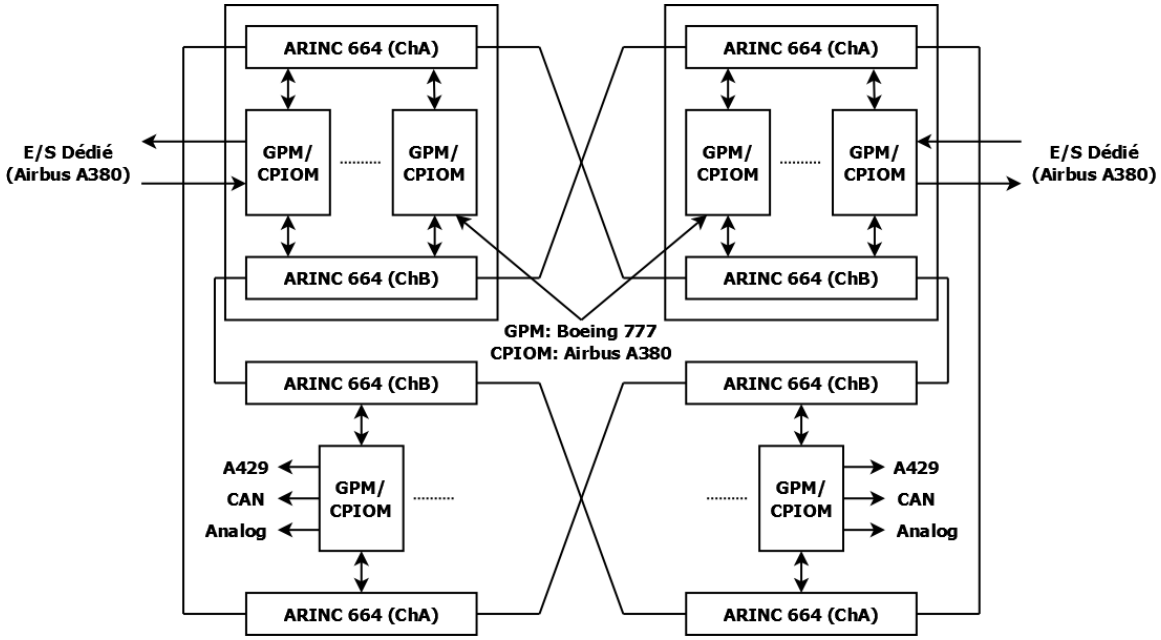


Figure 1-4: Exemple d'architecture IMA [1]

Ce nouveau type d'architecture est quant à lui basé sur une conception modulaire et un partage de ressources génériques entre plusieurs systèmes. Les bus mono-émetteur de l'approche fédérée sont entre autres remplacés par des bus multiplexés. Une normalisation plus poussée de l'architecture matérielle est donc requise pour une connexion harmonieuse à plusieurs systèmes. Au niveau applicatif, le développement peut s'effectuer de manière indépendante de la structure d'implémentation et doit également être en mesure de supporter plusieurs périphériques simultanément. Comme exemple d'application d'une architecture IMA dans la conception d'un avion moderne, nous pouvons citer le cas du A380 chez Airbus. L'approche IMA retenue par Airbus consiste à développer une série de composants de base pouvant être assemblés de manière modulaire pour répondre aux besoins de tout système. Dans le cas du A380, ces composants sont

utilisés pour construire les 30 modules du système incluant les modules requis pour l'atteinte des cibles de fiabilité grâce à la redondance. Par la suite, le fournisseur de chaque système développe le côté applicatif en respectant les interfaces des différents modules de base. Grâce à ce système, plusieurs applications (freinage, contrôle de vol, carburant, etc.) peuvent même se retrouver sur le même module tout en provenant de fournisseurs différents et complètement indépendants. La stratégie adoptée permet à Airbus dans ce cas d'améliorer l'efficacité de l'utilisation des ressources. L'avantage principal de l'utilisation de l'architecture IMA est donc de pouvoir optimiser les ressources utilisées contrairement aux modèles précédents.

1.2 Problématique

La plupart des concepts utilisés par l'architecture IMA, bien que déjà connus dans le domaine de l'informatique distribuée, constituent un changement marqué par rapport aux modèles antérieurs dans le monde avionique. Ceux-ci viennent s'ajouter aux contraintes importantes de l'avionique classique telles que le déterminisme, le temps réel, la certification et les cibles élevées de fiabilité. L'adoption de l'approche IMA a déclenché une révision de plusieurs aspects de la conception, de la certification et de l'implémentation d'un système IMA afin d'en tirer profit. Cette révision, ralentie par les contraintes avioniques, est toujours en cours, et offre encore l'opportunité de développement de nouveaux outils, méthodes et modèles à tous les niveaux du processus d'implémentation d'un système IMA.

Plusieurs aspects du design et de la validation des architectures IMA doivent être abordés de manière différente. Par exemple, l'utilisation d'un support matériel commun demande toujours une spécification propre, mais doit également être considérée dans les analyses de performance et de sûreté de chacun des sous-systèmes impliqués. L'interaction entre les différents systèmes doit être prise en compte afin de valider le comportement du système entier. Le développement d'applications évoluant sur le même support physique requiert également l'introduction d'une certaine forme de parallélisme pour l'intégration et la validation des différentes fonctions implémentées. Au niveau de la gestion des ressources, un partage global demande une gestion des ressources inexploitées au niveau système plutôt qu'au niveau local. Dans les architectures antérieures, la présence de ressources supplémentaires n'était nullement utile, puisque celles-ci ne peuvent être employées à accroître ou modifier les fonctionnalités des différents systèmes connexes. Dans une architecture IMA, la capacité de croissance est implicitement requise lors des

phases de conception et d'implémentation pour apporter une flexibilité au système et minimiser les efforts de modification.

Bien qu'au moins une vingtaine d'années se soient écoulées depuis l'introduction des principes de base des architectures IMA, il semble évident que les outils et méthodes permettant l'évaluation et la réalisation des phases de conception et d'implémentation n'ont pas évolué suffisamment pour permettre une utilisation optimale de ce type d'architectures. Les techniques et approches traditionnelles couramment utilisées seront présentées tout au long de cette thèse. Cette lente évolution des pratiques avioniques traditionnelles s'explique en partie du fait que les avancées technologiques aient été suffisantes du début des années 60 au tournant des années 2000 pour améliorer substantiellement les performances des systèmes avioniques. Les changements apportés par l'architecture IMA impliquent également une évolution des méthodes et outils de conception, d'implémentation et de validation pour continuer l'amélioration des performances de ces systèmes. Cette distinction peut, entre autres, expliquer le retard au niveau de l'évolution des outils et des approches, puisque l'architecture IMA demande une révision des méthodes.

Dans un contexte de proposition et de validation d'une nouvelle architecture IMA pour un réseau générique de capteurs à bord d'un avion, nous avons identifié quelques aspects des différentes approches traditionnelles pour la réalisation de ce type d'architectures pouvant être améliorés. Afin de pallier à certaines des différentes lacunes identifiées, nous avons proposé une approche de validation basée sur une plateforme matérielle reconfigurable ainsi qu'une nouvelle approche de gestion de la redondance pour l'atteinte des cibles de fiabilité. Contrairement aux outils statiques plus limités satisfaisant les besoins pour la conception d'une architecture fédérée, notre approche de validation est spécifiquement développée de manière à faciliter la conception d'une architecture IMA. De même, une application traditionnelle de la gestion de la redondance ne permettait pas de tirer profit de la flexibilité offerte par une architecture IMA. Les détails des différentes contributions de cette thèse seront présentés à la section 1.4.

1.3 Cadre de projet

Dans l'industrie avionique, de nombreux systèmes de capteurs et d'actuateurs sont requis afin d'assurer un nombre grandissant de fonctions à bord des avions. Avec les approches et outils conventionnels, la connexion de ces instruments demanderait un câblage réseau lourd et

encombrant. Les différentes cibles de fiabilité en avionique pour les systèmes critiques en complexifient le design de par l'installation redondante d'une grande partie des composants à bord d'un aéronef. De plus, la communication entre différents modules appartenant à divers domaines d'application est maintenant requise dans les nouvelles plateformes avioniques. Ceci entraîne une augmentation marquée de la demande en bande passante à l'intérieur des aéronefs, tout en devant s'assurer de respecter l'intégrité des données critiques lors de leur transmission.

Les travaux et propositions présentés dans cette thèse de doctorat ont été largement réalisés dans le cadre du projet AVIO402, qui tentait de répondre aux nouvelles demandes du domaine avionique énoncées précédemment. Le but de ce projet était donc de proposer de nouvelles architectures pour assurer des communications fiables de données critiques dans un avion en réduisant la complexité des systèmes câblés à l'aide de trois parties complémentaires :

- l'adoption de nouveaux bus de données multifonctionnels hautement fiables;
- le développement de nouvelles interfaces analogiques pour de nouveaux types de capteurs et actuateurs;
- l'exploration des effets des communications sans fil à bord d'un avion.

Une architecture appropriée a donc été développée en respectant chacun de ces aspects de manière à faciliter leur interopérabilité. Ces sujets sont des points d'intérêt reliés au développement technologique des systèmes avioniques de prochaine génération pour les partenaires industriels impliqués, soit Thales Canada inc. (Thales) et Bombardier Aerospace inc. (Bombardier). Ce domaine de recherche est également une des priorités principales du Consortium de Recherche et d'Innovation en Aérospatiale du Québec, le CRIAQ, qui a contribué financièrement à la réalisation de ce projet.

La nouvelle topologie qui a été développée dans le cadre du projet repose sur l'utilisation du protocole de transmission de données AFDX/ARINC664 [3] comme architecture de base du réseau. Cette norme permet une bande passante amplement suffisante pour les besoins en communication à bord d'un avion. Par contre, le coût élevé associé à chacune des interfaces de connexion à un réseau AFDX rend la connexion individuelle des capteurs/actuateurs relativement dispendieuse. Un réseau secondaire est donc nécessaire pour complémenter le réseau AFDX en regroupant une série de capteurs/actuateurs et éventuellement de plusieurs autres types de périphériques à un seul port d'accès AFDX. Le diagramme du système proposé est présenté à la

Figure 1-5. Afin de se conformer aux exigences en matière de fiabilité, le bus secondaire suit la norme ARINC 825 [4], une version modifiée du protocole CAN [5] qui répond spécifiquement aux besoins du domaine avionique.

Les diverses parties de ce réseau de capteurs ont été implémentées et validées sur une plateforme matérielle reconfigurable basée sur des FPGA afin de se rendre à un niveau de maturité technologique (Technology Readiness Level) TRL4, correspondant à une validation en laboratoire à l'aide d'un prototype. Cette plateforme matérielle servant de prototype a été utilisée pour l'évaluation des performances réelles de cette architecture en matière de bande passante et d'intégrité des données. Ce prototype comprend entre autres des mécanismes d'injection de pannes de manière à simuler différents types de défaillances et des modules matériels additionnels améliorant l'intégrité et la fiabilité du réseau.

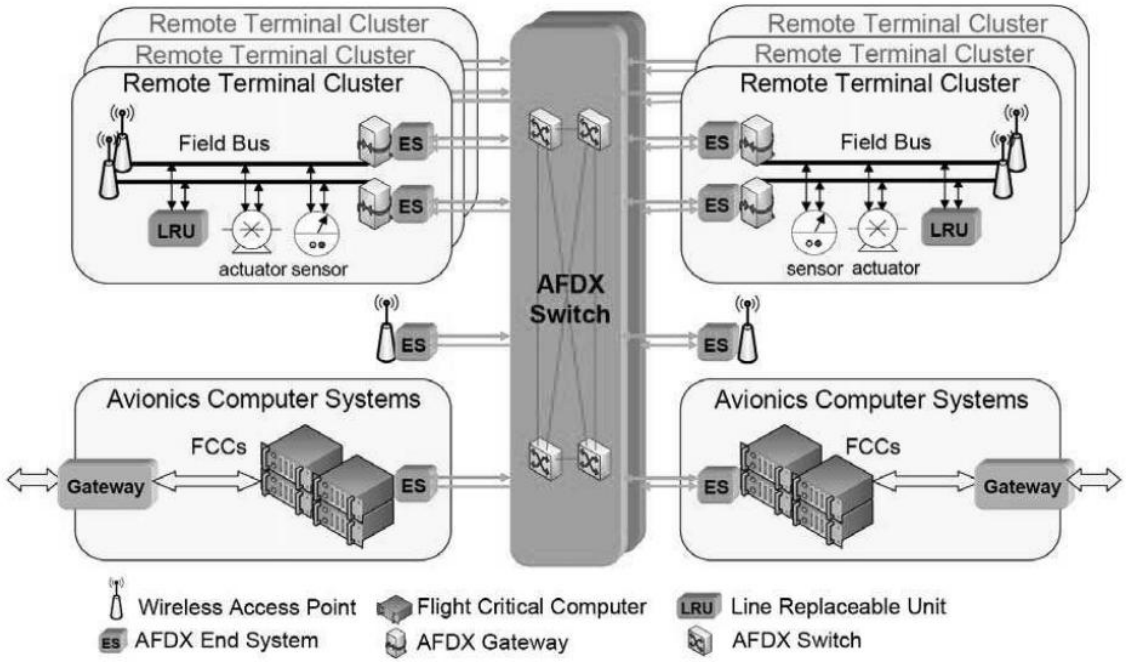


Figure 1-5: Architecture proposée [6]

Les travaux de cette thèse, qui s'inscrivent à l'intérieur du projet AVIO402, ont porté principalement sur l'élaboration d'une architecture de réseau de données plus efficace pour le réseau secondaire reliant les capteurs/actuateurs au réseau principal. Cette architecture se devait d'être la plus générique possible afin de desservir une quantité variable de capteurs/actuateurs de toutes sortes. Pour ce faire, l'architecture proposée s'est inspirée de la norme IEEE 1451 [7], qui suggère un modèle de référence standard permettant l'accès et le contrôle de transducteurs, d'une

manière la plus générique possible, à travers des interfaces communes vers tout système de contrôle ou d'instrumentation. La topologie proposée peut ultimement être implémentée avec une grande liberté quant au choix du protocole de transmission et des couches physiques du réseau, pouvant même supporter une éventuelle implémentation sans fil.

La fiabilité du réseau devait également être validée afin de satisfaire les niveaux requis imposés par le domaine avionique. Le réseau de transmission se devait d'avoir un taux de pannes inférieur à celui des composants qu'il connecte, soit les capteurs/transducteurs et les ordinateurs de vol. En raison de la nature critique des données recueillies par les capteurs et des commandes envoyées aux actuateurs pour le calcul et le maintien de la bonne trajectoire de vol, le niveau de tolérance aux pannes se devait d'être très élevé, soit de l'ordre d'une erreur non détectée ou d'une panne pour 10^9 heures de vol. Puisqu'un des buts de cette partie du projet AVIO402 était de proposer un réseau étant le plus générique possible, le degré de fiabilité devait également être ajusté en fonction du niveau requis pour l'application. Par contre, il est à noter que les hausses de fiabilité sont généralement accompagnées par une augmentation de la consommation de ressources matérielles de la plateforme matérielle, puisque cette hausse de fiabilité est traditionnellement acquise grâce à l'ajout de redondance dans le système. Une stratégie a été mise de l'avant pour ajuster la fiabilité tout en essayant de minimiser la consommation de ressources. Globalement, l'approche présentée cherche à mieux intégrer ces techniques de fiabilité afin de minimiser la consommation de ressources et l'effort de validation.

1.4 Objectifs de recherche et contributions

Dans le cadre de cette thèse, trois axes principaux de contributions originales se sont dégagés des travaux exécutés suivant les différents objectifs de recherche. Le premier objectif de recherche était de proposer une nouvelle architecture de réseaux avioniques pour la connexion de capteurs ou actuateurs. En effet, l'examen des architectures utilisées traditionnellement dans l'industrie ne permettait pas l'atteinte des requis fixés en matière de flexibilité, reconfigurabilité et tolérance aux pannes. Afin de pouvoir répondre aux besoins imposés par le cadre du projet AVIO 402, la proposition d'une nouvelle architecture a donc été nécessaire. Cette architecture fait l'objet d'un premier article [8] intitulé « A System Architecture for Smart Sensors Integration in Avionics Applications » et publié dans la revue « SAE International Journal of Aerospace » en 2012. La

version intégrale de cet article se trouve au chapitre 3. Les contributions de cette architecture de systèmes avioniques sont :

- Afin de conférer un caractère générique et configurable aux architectures avionique existantes, une nouvelle architecture hiérarchique est proposée. En se basant entre autres sur la norme IEEE 1451, l'architecture proposée permet la connexion des capteurs et actuateurs intelligents à tout système numérique.
- L'architecture hiérarchique innove en permettant une optimisation des ressources consommées et de la fiabilité selon les contraintes à considérer. Un système critique peut donc être implémenté de manière à assurer un niveau élevé de fiabilité tandis que l'on peut minimiser la consommation de ressources d'un système non critique grâce à la même architecture hiérarchique.
- En comparaison avec des architectures similaires suivant la norme 1451, nous avons pu établir que pour un même niveau de fiabilité, l'architecture hiérarchique proposée entraîne une réduction de la consommation de ressources quand on la compare avec des architectures existantes.

Le second objectif de recherche visait la proposition d'une nouvelle approche de conception et de validation adaptée pour l'implémentation de réseaux avioniques suivant les principes de l'IMA. Pour faire suite à la problématique exposée à la section 1.2, nous avons en effet établi que les outils de conception et de validation utilisés pour les architectures précédant l'introduction de l'IMA n'étaient toujours pas adaptés à nos objectifs de travail. La nouvelle approche basée sur une plateforme matérielle a fait l'objet de la section 5.2. Les contributions de cette approche matérielle sont :

- Une approche basée sur l'exploitation d'une plateforme matérielle servant à la phase initiale de conception qui prend en considération les principes IMA et permet la validation de l'architecture sélectionnée au niveau approprié. Les approches traditionnelles logicielles se concentrent sur la validation d'un simple aspect de l'architecture plutôt que de la considérer dans son ensemble.
- Afin de pouvoir faire une validation matérielle multiplateforme de notre architecture, le développement d'un nouveau mécanisme hybride de mesure de latence des trames du

réseau a été nécessaire. En effet, les moniteurs existants ne permettaient pas une mesure de latence du réseau dans son entier en ne supportant l'examen que de sous-sections.

Dans le cadre du processus de validation de notre architecture hiérarchique à travers la nouvelle approche matérielle, un troisième objectif de recherche, exposé au chapitre 5.2, prenait la forme d'une évaluation de possibles améliorations pouvant être apportées au protocole ARINC 825. Les contributions au niveau des améliorations proposées sont :

- L'inclusion dans les contrôleurs ARINC 825 de stratégies d'agrégation de trames qui permettent la réduction de la bande passante consommée et la connexion d'un nombre supérieur de capteurs ou actuateurs pour la même configuration d'architecture hiérarchique.
- La modification au protocole de gestion d'erreur des contrôleurs ARINC 825 qui a été nécessaire afin d'assurer le fonctionnement correct des mécanismes de tolérance aux pannes.

Le dernier objectif de recherche cherchait à déterminer la méthode appropriée d'introduction de la redondance dans un réseau avionique afin d'atteindre des cibles de fiabilité. Notre approche à la gestion de la redondance pour les systèmes avioniques est présentée dans un article [9] intitulé « Hierarchical Redundancy Management for Avionic Networks » soumis à la revue « Transactions on Aerospace and Electronic Systems ». Cet article se trouve au chapitre 5. Les contributions au niveau des améliorations proposées sont que:

- L'analyse de l'architecture hiérarchique révèle qu'elle permet de réduire les effets de couplages indésirables en avionique en réduisant le nombre de points communs de défaillance, contrairement aux approches traditionnelles d'introduction de redondance.
- Dans le cadre d'introduction de redondance dans un réseau avionique, l'approche hiérarchique permet également de réaliser un gain au niveau de la consommation de ressources comparativement aux techniques usuelles de gestion de la redondance.

1.5 Organisation de la thèse

L'introduction de cette thèse a présenté l'évolution du domaine avionique au niveau des architectures de réseau ainsi que les différentes problématiques qui s'y rattachent. À partir du cadre de travail des réalisations effectuées, les différentes contributions réclamées dans cette

thèse ont été énoncées. Le reste de la thèse continue selon la structure suivante. Le chapitre 2 explore la littérature pertinente et aura pour but de présenter et justifier les techniques et approches ayant déjà prouvé leur efficacité dans un passé rapproché, ainsi qu‟à établir les principes de base dans les domaines pertinents. Ce chapitre est divisé en 3 sections rattachées à chacun des 3 axes de contribution afin de les mettre plus spécifiquement en contexte.

Deux articles pertinents à ces contributions constituent la suite de cette thèse. Le premier article, composant le chapitre 3, présente l‟architecture générique ainsi que ses différentes distinctions et avantages d‟utilisation. Le chapitre 4 contient le dernier article portant sur la gestion de la redondance de l‟architecture hiérarchique ainsi que sur l‟analyse des points communs de défaillance pour les différentes méthodes étudiées. Le chapitre 5 contient des détails additionnels de nos travaux de recherche en faisant un retour sur les articles exposés sous la forme d‟une discussion générale. Les différents choix menant aux résultats sont également discutés. Finalement, le chapitre 6 vient conclure cette thèse en résumant brièvement son contenu et en donnant quelques indications sur les possibles améliorations futures.

CHAPITRE 2 RÉSEAUX AVIONIQUES DE CAPTEURS

Quand on analyse les premières générations de systèmes avioniques antérieurs à l'avènement du concept IMA au tournant des années 2000, on constate que l'évolution des architectures se résumait principalement à une simple évolution des composants rendus désuets par l'avancement des technologies numériques. Avec l'arrivée de l'IMA, plusieurs aspects des méthodes de design et validation ont dû être adaptés pour faire face aux nouvelles possibilités. Pour ce faire, les connaissances de plusieurs autres domaines (temps réel, fiabilité, capteurs intelligents, etc.) ont été intégrées aux techniques traditionnelles afin de faire face à ces nouvelles problématiques, tel le partage de systèmes génériques, les interactions entre différentes applications et la reconfiguration dynamique, spécialement dans le cadre d'applications critiques. Les principales contraintes au niveau d'applications critiques seront présentées dans la section portant sur les méthodes de design et validation de systèmes avioniques.

Le présent chapitre met en contexte chacun des articles composants cette thèse en présentant de manière plus spécifique les principaux champs d'intérêt et technologies abordés. Le champ d'intérêt de la première contribution évoquée dans [8] couvre l'intégration du protocole IEEE 1451 [7] au processus de design de systèmes avioniques. Pour débuter, les principes de base de la norme IEEE 1451 seront donc exposés afin de mettre en contexte les contributions contenues dans l'article [8]. Les différentes applications de la norme IEEE 1451 ainsi que les architectures résultantes de son intégration feront par la suite ressortir les avantages et limites de la transposition de ces différentes architectures dans le domaine avionique. Les différentes tendances et derniers avancements significatifs seront finalement exposés pour mettre en lumière l'intérêt de proposer une nouvelle architecture intégrant le protocole IEEE 1451 tout en respectant les spécificités du domaine avionique.

Les contributions énoncées de cette thèse se situent au niveau des différentes approches de design et validation de réseaux avioniques. L'approche classique de design sera d'abord présentée en complétant avec les différentes variantes les plus pertinentes. Les différentes stratégies décrites et classées par type d'approche logicielle ou matérielle viendront donc s'ajouter à celle de ce dernier article. La dernière partie de cette section présentera les derniers travaux dans le domaine des moniteurs de surveillance pour situer une partie des contributions évoqué au chapitre 5.2. Le protocole ARINC 825, incluant l'implémentation d'un nœud de contrôle, sert de base aux

prochaines contributions énoncées dans cette thèse. Les principes de base du protocole ARINC 825 seront donc exposés ainsi que l'état de la recherche récente sur cette norme. Cette section se conclura par la présentation des stratégies de regroupement de trames en faisant ressortir les avantages et inconvénients de l'utilisation de ces techniques. Les derniers avancements dans le domaine seront également présentés.

Le dernier thème abordé dans cette revue de littérature portera sur la gestion de la redondance pour mieux mettre en lumière les contributions du dernier article [9] composant cette thèse. L'approche classique d'intégration de redondance dans un système avionique sera d'abord présentée. L'architecture Com-Mon sera exposée à ce niveau afin de venir complémenter les architectures de comparaison déjà incluses dans le second article [9]. Les derniers travaux pertinents présentés depuis la publication de [9] seront par la suite présentés en focalisant sur l'importance du système de votation lors de l'utilisation de la redondance. Pour compléter, une courte liste des approches alternatives à la redondance sera également exposée.

2.1 IEEE 1451

Pour commencer, une courte explication présentera les principes de base de la norme IEEE 1451 suivie par les principales applications et architectures découlant de la recherche dans le domaine de ce protocole. De plus amples détails sont inclus dans l'article du chapitre 3. Les références contenues dans cette thèse reflèteront donc les avancements réalisés et les tendances observées depuis 2012, soit la date de parution du premier article. Toutes les références pertinentes publiées antérieurement peuvent être trouvées directement dans l'article du chapitre 3.

2.1.1 Principes de base

Les différents types de capteurs et d'actuateurs sont une partie importante de plusieurs systèmes dans les domaines de la production en chaîne, du contrôle industriel, des transports, de l'aéronautique, etc. Puisque ceux-ci sont utilisés dans de nombreuses applications, leurs implémentations sont généralement spécifiques à une classe précise d'applications. Le support de plusieurs protocoles de communication et de réseaux ou le développement de plusieurs versions du même transducteur demanderait des efforts significatifs qui s'avèrent peu avantageux pour le manufacturier. La norme intitulée « *IEEE 1451 : Standard for a Smart Transducer Interface for Sensors and Actuators* » [7] a été proposée afin de régler ces problèmes. Son objectif principal est

de proposer un cadre d'implémentation pour le contrôle de transducteurs à travers des interfaces normalisées. En suivant la norme, les transducteurs peuvent donc être connectés directement à un ordinateur, un système d'instrumentation ou tout autre type de réseau de contrôle. Le modèle de référence mis de l'avant est complètement indépendant du protocole de communication et même du choix d'une implémentation logicielle ou matérielle.

Le respect de cette norme permet de diminuer l'effort de design, en particulier lors d'utilisations dans de multiples applications aux caractéristiques différentes. Le système devient également plus facile à installer, mettre à jour, remplacer et déplacer. Ces derniers avantages sont particulièrement recherchés lors de la sélection de tout système inclus à bord d'un aéronef.

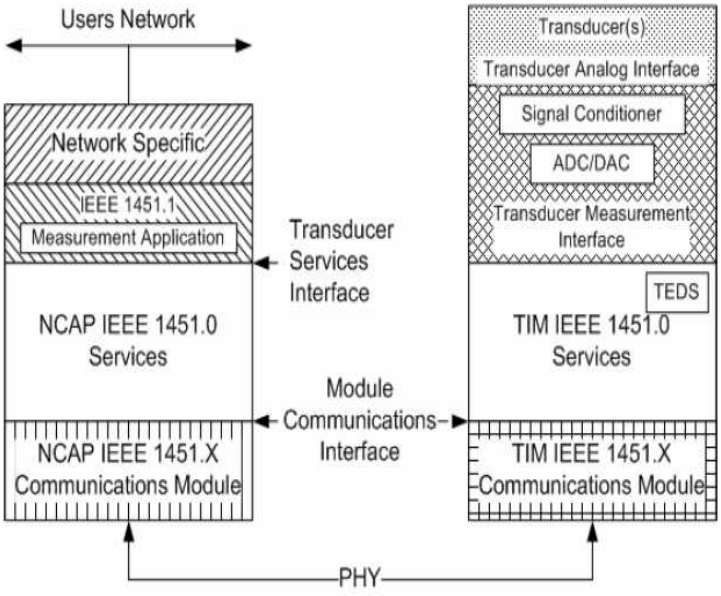


Figure 2-1: Modèle de référence du IEEE 1451 [7]

Le modèle de référence de la norme IEEE 1451 est présenté à la Figure 2-1. La section centrale de la spécification propose une série d'opérations et de commandes séparées en deux modules principaux. Le « Transducer Interface Module (TIM) » est responsable du conditionnement du signal analogique, de la conversion numérique, du traitement des commandes, du transfert des données et du stockage des informations relatives aux transducteurs dans un module appelé « Transducers Electronic Data Sheet (TEDS) ». Chaque TIM contient également au moins un transducteur tandis que les TEDS doivent être contenus dans une mémoire non volatile. La deuxième partie, le « Network Capable Application Processor (NCAP) », est responsable de lier

les modules TIM au réseau principal. Les NCAP sont responsables du contrôle des interfaces, de la correction des données et de l'interprétation et encodage des messages. Plus de détails sur l'IEEE 1451 sont inclus dans les deux premiers articles [8, 10] de cette thèse.

2.1.2 Applications

Une première revue de littérature sur les usages possibles du IEEE 1451 [11], publiée en 2015, montre bien l'état présent de la recherche dans ce domaine. De manière globale, aucune application n'a nécessité de modifications majeures de la norme depuis son introduction en 1997. Les travaux réalisés ont en majorité eu comme objectif principal de s'assurer l'applicabilité de la norme dans des domaines précis, afin d'ajouter au domaine un caractère générique et flexible. Pour en arriver à une intégration réussie, de nombreuses interprétations mineures se conformant toujours à la norme de base ont été proposées. Notre approche hiérarchique constitue à nos yeux la première proposition architecturale apportant une contribution substantielle à la norme en y conférant un caractère fiable. En effet, l'introduction de mesure architecturale de gestion de la redondance globale, afin d'assurer un niveau adéquat de fiabilité, est requise pour une utilisation en avionique dans le cadre d'applications critiques. Par pallier à ce manque de la norme originale, nous proposons donc dans [8] notre architecture hiérarchique permettant un niveau modulable de fiabilité. Malgré l'ajout de la fiabilité à la norme, notre architecture hiérarchique s'y conforme toujours grâce aux différentes libertés d'implémentation permises. Plus de détails et d'analyses à ce sujet suivront la présentation des 3 articles à la section discussion. Afin d'illustrer ces derniers propos, le reste de la section présentera les exemples les plus pertinents afin de faire ressortir la diversité des champs d'application et des technologies de communication utilisées pour mieux situer nos travaux.

Un premier cas typique de modifications apportées au protocole IEEE 1451 pour répondre aux besoins de nouveaux domaines d'application est proposé par Grisotomi et al. [12]. Leur intégration du IEEE 1451 dans un contexte de surveillance de procédé industriel permet la diversification de l'approvisionnement et une facilité accrue d'entretien, causant une augmentation de la fiabilité et de l'efficience globale. Au niveau de la connexion entre TIM et NCAP, l'implémentation à l'aide de microcontrôleurs repose sur le protocole sans fil IEEE 802.15.4 [13] issu de la famille des « Low Rate Wireless Personal Area Network » (LR WPAN). Leur approche modulaire, scindant le TIM en deux, leur permet d'accroître la polyvalence de

l'implémentation en fonction de requis spécifiques au domaine d'exploitation. Malgré leur architecture, présentée à la Figure 2-2, l'implémentation respecte toujours la norme IEEE 1451.

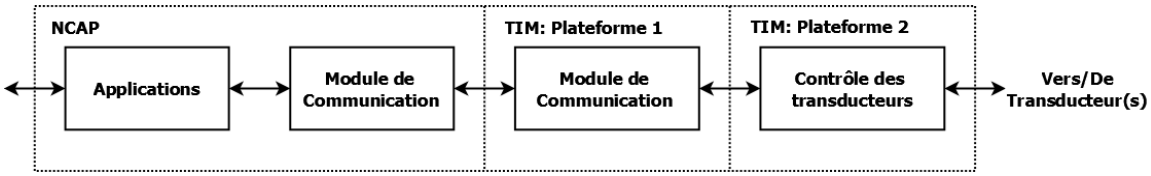


Figure 2-2: Architecture réseau [12]

Cette séparation du TIM, n'étant pas spécifiée par la norme, leur donne la liberté de séparer physiquement en deux modules distincts le contrôle du réseau sans fils et du transducteur. Cette division du TIM a également entraîné la répartition de l'information des TEDS sur les deux plateformes, un premier pour l'adressage et la connexion au réseau et le second pour le contrôle des transducteurs. Aucune nouvelle fonctionnalité, comme de nouvelles informations à ajouter aux TEDS ou un style d'adressage différent, n'a par contre été requise. On peut noter que malgré l'introduction de cette séparation, le type d'architecture se classe parmi le style de base selon la classification introduite dans notre article. Une deuxième proposition distincte ressort du travail de Bezanic et als. [14] sur leur approche pour faciliter l'intégration de nouvelles fonctionnalités dans une architecture respectant la norme IEEE 1451. Ces nouvelles fonctionnalités sont intégrées à des transducteurs virtuels qui sont considérés pareillement aux transducteurs physiques au niveau réseau. Parallèlement à l'intégration de services web à leur implémentation IEEE 1451, Bezanic et als. [14] se servent de ce nouveau concept de transducteur virtuel pour améliorer leur design d'un système de dégivrage. L'utilisation de services web connait présentement une croissance importante dans les domaines de surveillance, de l'environnement et des transports intelligents. L'architecture proposée par [14], présentée à la Figure 2-3, constitue donc le premier effort d'intégration du IEEE 1451 dans un système implémenté à l'aide du « Web Service Description Langage » (WSDL). Le code source WSDL est compilé et programmé sur des plateformes matérielles développées autour d'un processeur embarqué ARM.

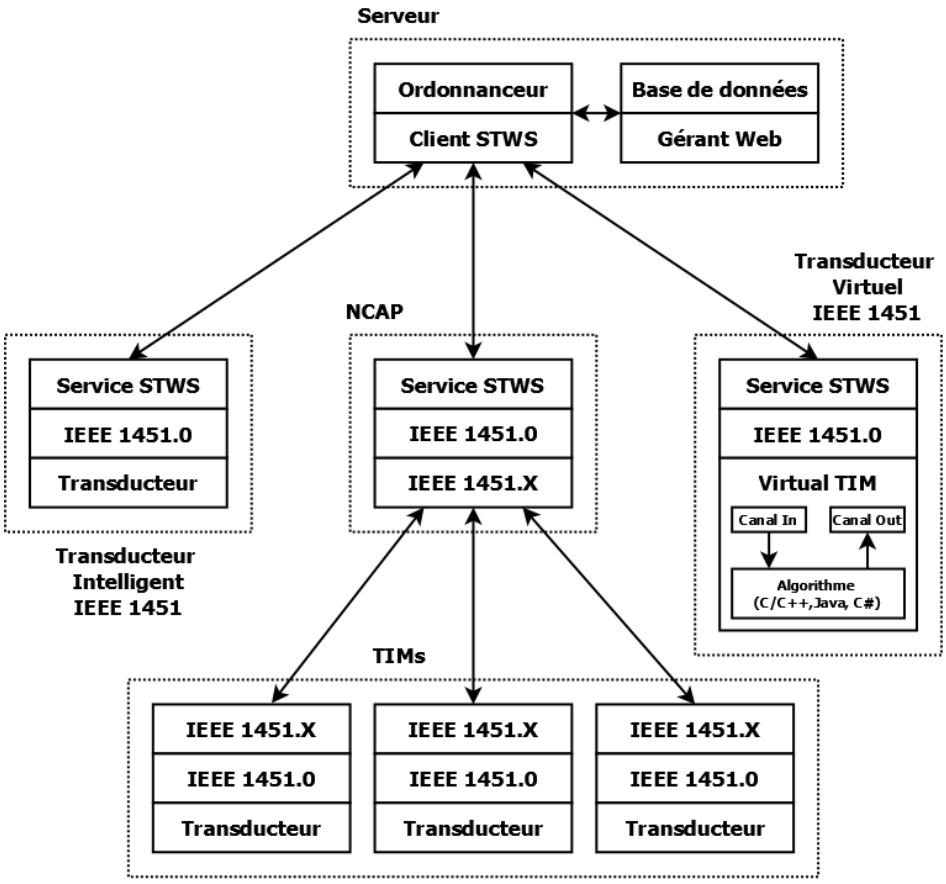


Figure 2-3: Architecture réseau [14]

Bien que le TIM virtuel possède les canaux d'entrée et de sortie, telle sa version réelle, un algorithme contenu dans le TIM virtuel traite les données entrantes et produit les données à transmettre. Dans le cas exposé par [14], un algorithme de prédiction de la formation de glace apporte des données supplémentaires au serveur de contrôle gérant l'activation des dégivreurs. Par l'entremise du réseau sans fil, le transducteur virtuel achemine des informations supplémentaires au serveur de contrôle à l'aide des données brut des capteurs. Le transducteur virtuel permet donc dans ce cas d'implémenter un algorithme au niveau des transducteurs plutôt que dans le serveur. Bien que les architectures générées respectent toujours la norme IEEE 1451, le design et l'emploi du transducteur virtuel ne sont pas balisés par la norme. Au niveau architectural, la structure obtenue peut supporter les deux types de topologies, soit celle de base et la distribuée. Le tableau suivant vient compléter l'analyse des travaux pertinents dans le domaine du IEEE 1451 à l'aide d'un résumé de leurs principales caractéristiques. Chacune des références couvre donc une intégration du IEEE 1451 dans le domaine d'application indiqué.

Tableau 2.1: Domaines d‟application de la norme IEEE 1451

Référence	Année de publication	Domaine d'application	Méthode d'implémentation	Architecture	Protocole de communication
12	2014	Procédé industriel	Microcontrôleur	Base	IEEE 802.15.4
14	2015	Dégivrage	Services Web	Distribué	Ethernet
15	2013	Protection de l'environnement	SoC	Base	Dédié
16	2014	"Internet of Things" (IOT)	Simulation Logicielle	Distribué	Non Applicable
17	2015	Qualité de l'eau	C/C++	Distribué	Nordic Multiceiver
18	2014	Système de transport intelligent	Visioway (IP)	Distribué	Wi-Fi
19	2014	Aerospace	Non Disponible	Distribué	Zigbee
20	2013	Validation électronique	FGPA	Base	UART

On peut voir qu‟au cours des dernières années, le protocole IEEE 1451 a été sélectionné et implémenté avec succès dans de nombreuses nouvelles sphères d‟activités. Au niveau avionique, il est important de noter que bien que les travaux de Alena et al. [15] aient été effectués pour le compte de la NASA, leur utilisation du IEEE 1451 n‟était limitée qu‟à la reconfiguration des modules et aux tests au sol. L‟architecture développée n‟était donc pas assujettie aux règles de l‟avionique, mais plutôt à celle du domaine de l‟instrumentation et de la programmation. Suite à cette revue extensive des travaux dans le domaine du IEEE 1451, nous pensons être en mesure d‟affirmer que nous sommes les premiers chercheurs à introduire le IEEE 1451 dans le cadre d‟applications critiques en avioniques grâce à notre proposition d‟architecture pouvant entre autres conférer le niveau requis de fiabilité.

Au niveau de la méthode d‟implémentation, on peut conclure que les approches logicielles et matérielles ont été également privilégiées. Certains [16] ont même choisi d‟implémenter le réseau de contrôle des autobus d‟une ville entière à l‟aide d‟un système de capteurs propriétaire (Visioway). Malgré toutes les applications recensées, aucun processeur spécialisé n‟a été proposé jusqu'à maintenant. Bien que le type d‟architecture dépende beaucoup du domaine d‟application,

les architectures distribuées tendent à être plus fortement favorisées, puisqu‟elles permettent la connexion d‟un plus grand nombre de transducteurs afin de maximiser les capacités des ressources matérielle et logicielle. Les augmentations de bande passante dues aux protocoles supportés viennent accentuer cette situation en permettant le regroupement d‟un plus grand nombre de transducteurs sous un même réseau. Les nombreux protocoles de communication en usage dans les travaux du tableau 1 montrent bien la polyvalence de la norme IEEE 1451, un de ses fondements de base. Les choix varient de protocole dédié [17] conçu pour une application unique jusqu‟aux réseaux sans fils [12, 15, 16] en utilisation grandissante. Au niveau des protocoles de communication avioniques, aucune référence formelle n‟a été trouvée parmi la littérature examinée. De nombreux articles font par contre référence au protocole de transmission CAN pour application non critique, mais aucune mention de l‟ARINC 825 en combinaison avec le IEEE 1451 n‟a été trouvée, jusqu‟à la parution de notre article [8].

Globalement, nous pouvons donc conclure qu‟au niveau de la recherche sur le IEEE 1451, les dernières avancées se résument principalement à de simples ajustements afin d‟accommoder les particularités spécifiques de chacun des nouveaux domaines d‟application considérés. Notre approche présentée dans [8] implique une évolution majeure du IEEE 1451 afin de satisfaire la fiabilité requise dans les applications critiques.

2.2 Design et validation de systèmes IMA

Pour commencer, une courte explication présentera l‟approche classique guidant le design et la validation de systèmes avioniques depuis l‟introduction des IMA. Les références composant cette section reflète donc les derniers avancements réalisés et les tendances du domaine.

2.2.1 Approche classique

De nos jours, la méthodologie classique de design de systèmes IMA reprend les techniques utilisées pour les paradigmes de développement antérieurs applicables aux systèmes avioniques en incorporant les nouvelles particularités des architectures IMA. En raison de la nature configurable des architectures IMA, les différentes ressources doivent être assignées aux multiples fonctionnalités selon le temps alloué, la quantité de mémoire requise, les entrées/sorties des réseaux de communication et de l‟interfaçage avec les périphériques externes. L‟allocation des ressources s‟effectue habituellement à l‟aide de tables de configuration précisant les

différentes propriétés à respecter pour chacune des fonctions à implémenter. Ces propriétés permettent la modification individuelle d'une fonction sans affecter le système dans son entier. La définition de ces propriétés sert également de base pour les nombreux processus de certification. Globalement, la méthode de design IMA suit les étapes suivantes :

1. Définition des fonctionnalités desservies par le système;
    - Identification des besoins en entrées/sorties et des périphériques externes tels que capteurs et actuateurs;
2. Définition des éléments de traitement requis pour l'implémentation de chacune des fonctions;
    - Proposition d'une architecture satisfaisant les requis de fiabilité et de performances;
3. Définition des besoins en ressources pour chacune des fonctionnalités;
4. Définition des échanges entre les éléments de traitement et les fonctionnalités;
5. Allocation des éléments de traitement à la plateforme physique.

L'allocation prédéterminée des ressources est finalement communiquée à la plateforme physique à l'aide d'un fichier de configuration. Les configurations obtenues devraient donc s'assurer du respect des différentes propriétés énoncées lors de la première étape. Le processus de design de systèmes avioniques est également sujet à de nombreux requis émanant de règlementations en vigueur et des spécifications. Afin de bien montrer la quantité imposante requise d'analyses de toutes sortes, la Figure 2-4 indique les relations entre les principales normes à respecter. Les guides « Avionics Recommended Practices » (ARP) sont publiés par la Society of Automotive Engineers (SAE) depuis 1989 et leur respect est toujours obligatoire à ce jour.

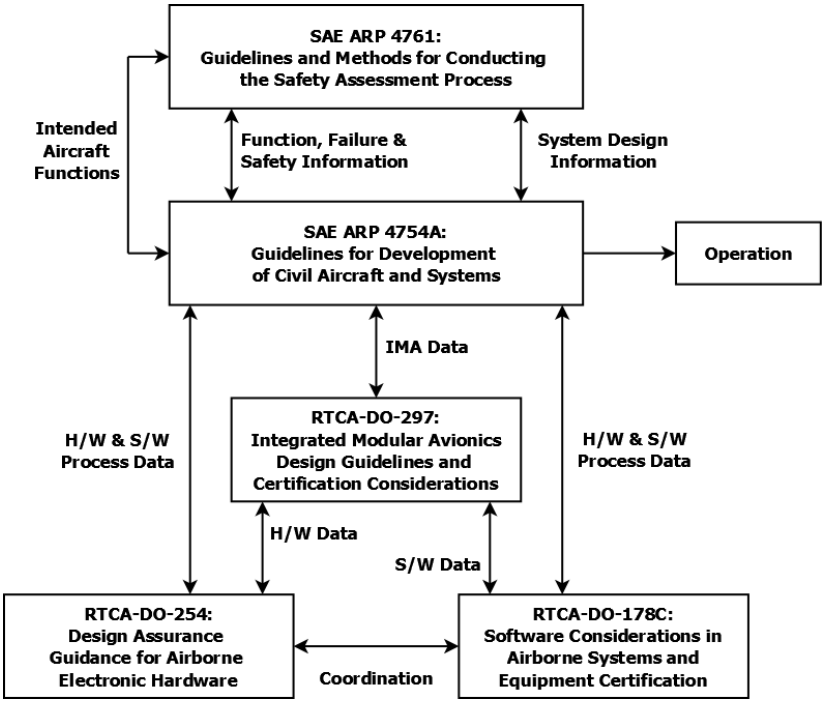


Figure 2-4: Relation entre les principales normes avioniques [1]

Le guide ARP 4754A contient les détails globaux du processus de design d'un système avionique en faisant, entre autres, le lien entre les différents processus et la documentation, tel qu'illustré à la Figure 2-4. Le document ARP 4761 détaille, quant à lui, toutes les cibles et analyses de fiabilité, incluant les « Fault Tree Analysis » (FTA), les Failure Modes and Effect Analysis » (FMEA) et les analyses de Markov. Ces derniers outils ont tous été utilisés par les membres de l'équipe AVIO 402 afin d'assurer le respect des normes avioniques dans le design et la validation de notre prototype. Ces analyses font également l'objet de recherches tentant entre autres de faciliter leur application sur des réseaux [18].

De manière plus spécifique aux systèmes IMA, la norme DO-297 en présente les fondements en précisant les lignes directrices permettant son développement et son intégration. Pour terminer, les deux derniers guides décrivent en détail les processus de certification requis. Le document DO-178 cible la programmation d'applications logicielles, tandis que le développement matériel est visé par le guide DO 254. Plusieurs autres normes peuvent venir s'ajouter à celle présente dans la figure 9, telles les normes ARINC qui varient en fonction des choix de protocoles et d'implémentations. Les contraintes utilisées dans le cadre de ce projet proviennent donc de ces

documents. Leur conformité a également été vérifiée par différents intervenants du domaine avionique pour un respect des conditions réelles d'exécution.

2.2.2 Approches originales

Afin de satisfaire tous les concepts et règles contenues dans les derniers documents présentés, de nombreux travaux ont cherché à proposer des variantes ou des améliorations des processus établis. Les sections suivantes viennent brosser un portrait global des travaux les plus récents. Chaque section est consacrée à une des deux classes d'approches de design et validation, soit l'approche basée sur la modélisation logicielle et les méthodes reposant sur l'utilisation directe de plateforme matérielle. Les particularités des deux approches principales recensées seront brièvement décrites afin de mieux situer les travaux pertinents dans le même champ d'activités depuis la parution de notre deuxième article. [10] Les travaux présentés dans cette thèse viendront donc appuyer les références incluses dans cette seconde publication.

2.2.2.1 Modélisation logicielle

La modélisation logicielle de systèmes est utilisée largement dans le processus de design et de validation des systèmes avioniques. Habituellement basée sur un ensemble mathématique approprié, l'analyse d'un modèle permet la validation du système à l'aide d'outils automatisés afin de détecter le plus rapidement possible les anomalies tout en identifiant les aspects du système à réviser. Les travaux effectués dans le cadre du projet MultiPARTES [19] visant le développement de systèmes à criticité mixte sur une plateforme multicoeur se basent entièrement sur une approche de modélisation. Leur approche repose sur la génération et l'analyse de 3 modèles distincts basés sur le langage UML2. Un premier modèle de l'application représente la description fonctionnelle du système incluant également des contraintes additionnelles de criticité et de synchronisation. Un deuxième modèle vient représenter la plateforme matérielle tandis que le dernier vient modéliser la relation entre les deux premiers modèles tout en spécifiant les contraintes de partitionnement. Ils obtiennent ainsi un partitionnement de leur application sur leur plateforme matérielle au prix de l'élaboration de 3 modèles complexes couvrant le système dans son entier. On peut voir dans ces derniers travaux la grande complexité de leurs étapes de modélisation tout en ayant ultimement besoin de la plateforme matérielle pour la validation finale.

Plutôt que de générer un modèle complexe à partir de zéro, l'outil de modélisation OPNET [20] est utilisé par plusieurs pour accélérer le processus de modélisation. Depuis son introduction en 1987, OPNET est devenu l'outil le plus utilisé pour les concepteurs de réseaux [21] avec sa capacité de rapidement modéliser tout type de trames, commutateurs, routeurs, terminaux, etc. À l'aide d'une modélisation basée sur le « Network Calculus » exécutée sur la plateforme OPNET, Zhao et al. [22] ont tenté de déterminer les délais maximaux encourus par des trames dans un réseau AFDX en fonction de plusieurs stratégies d'ordonnancement au niveau des commutateurs. À travers leurs travaux, les auteurs ont démontré la pertinence de leur modèle mathématique, bien que leur modèle reste limité aux délais dans un réseau AFDX. De manière similaire, Safwat et al. [23] proposent également un modèle de commutateur AFDX développé à l'aide d'OPNET. Finalement, Louati et al. [24] utilisent quant à eux la modélisation de réseaux de Petri à l'aide d'UML pour la validation de contraintes temporelles. Dans les quatre dernières approches énumérées, les méthodes originales ne ciblaient que les performances des réseaux avioniques sans se soucier des autres aspects du processus global, telle la validation du respect des cibles de fiabilité, contrairement à notre approche mixte qui considère tous ces processus dans leur ensemble. D'autres membres de l'équipe du projet AVIO402 ont également proposé une approche globale reliant la validation des contraintes de performances à celle de fiabilité. Li et al. [25] propose une méthode de modélisation de violation d'échéance de trame dans un réseau AFDX ne se basant plus sur des bornes pessimistes et permettant l'inclusion du mécanisme de redondance intégré. Employée en conjoncture avec une analyse FTA conventionnelle, la modélisation proposée permet l'inclusion de ces violations des performances dans les analyses de fiabilité, parallèlement à l'approche proposée dans cette thèse unifiant les analyses de fiabilité et de performances sous un processus de développement global. Li et al. [26] ont également été les premiers à considérer dans leur analyse de fiabilité l'effet de l'inversion de séquence de la norme AFDX qui s'avère être déterminant dans le niveau de fiabilité du protocole AFDX. L'effet sur la performance et la fiabilité de leurs deux méthodes de mitigation proposées ont pu être validé grâce à la modélisation présentée en [25].

Plusieurs travaux se sont quant à eux concentrés sur la modélisation de la fiabilité. Huan et al. [27] ont proposé leur propre langage afin de formaliser les différentes contraintes du système étudié. Leur approche est validée avec l'élaboration d'un système sujet aux contraintes imposées par la norme DO-178C. Après l'analyse automatique de leur modèle, les différentes contraintes

ne pouvant être respectées sont identifiées. Grâce à leur outil, Huan et al. ont pu prouver l'applicabilité du DO-178C dans leur design et ainsi proposer une amélioration d'une des parties du processus de design de système avionique. Notre approche prévoit plutôt l'inclusion des différentes étapes de certification directement dans le processus global, puisque notre approche se base, entre autres, sur les implémentations projetées qui doivent prendre en considération les contraintes réelles. Contrairement à [27], nous n'avons pas d'étape supplémentaire à effectuer pour s'assurer du respect du DO-178C, puisqu'il ferait parti du processus depuis le début. Afin de mieux capturer les modes de défaillances, Garro et al. [28] ont inclus des modèles de comportements dysfonctionnels dans leur méthode de modélisation RAMSAS permettant une prédiction plus juste du calcul de fiabilité de leur système. Cette méthode [29], elle-même récemment considérée pour le design de systèmes avioniques, se base entre autres sur les outils d'analyse de fiabilité contenus dans les guides ARP, tels les FTA et FMEA. Ces derniers travaux montrent encore que la plupart des propositions ciblent un aspect limité du processus global, puisque, dans ces derniers cas, le respect des contraintes de performance n'est pas pris en compte. Il est à noter que notre approche mixte tente de prendre en compte tous les aspects du design de systèmes avionique en assurant leur validation finale en respect entre autres avec les contraintes de certification. De manière globale dans l'approche logicielle basée sur des modèles, la complexité des modèles croît avec la complexité du système. Un effort important est donc requis afin de modéliser le plus précisément le système en développement. Un écart trop grand entre le modèle logiciel et le comportement réel peut mener à des résultats peu fiables. Cette dernière considération importante peut par contre être renforcée lorsqu'utilisée en tandem avec la plateforme mixte présentée au chapitre 5.2.

2.2.2.2 Plateforme matérielle

Au niveau de l'utilisation de la plateforme matérielle dans le processus de design et validation de systèmes avioniques, [30] commence par reconnaître la grande diversité des différentes approches. En effet, la plupart des approches reposent sur des langages d'implémentation dédiées pouvant difficilement répondre aux besoins grandissants du domaine. Afin de pallier à cette hétérogénéité des langages et plateformes, [30] propose un retour vers une approche combinant une part importante de modélisation. Suite à une modélisation du système, une traduction automatique vers le langage approprié permettrait donc de réutiliser le modèle de base pour

d‟autres applications ou pour tous changements apportés. Malgré l‟utilisation d‟une plateforme matérielle dans la boucle, la proposition de [30] implique quand même le recours à une phase de modélisation importante, ce qui s‟avère toutefois toujours nécessaire puisque certaines analyses ne peuvent être réalisées par expérimentation directe. Les analyses de fiabilité basées sur les taux d‟échec tel qu‟imposé par les différentes normes sont limitées au domaine logiciel, étant donné qu‟il n‟est généralement pas pratique d‟observer des taux d‟échec de composants réels. Notre approche tente, par contre, de maximiser la validation au niveau matériel en ajoutant des outils de monitorage pouvant extraire les caractéristiques du trafic ou injecter des pannes. Il est à noter que les outils de modélisation logicielle peuvent parfois servir directement d‟outil de monitorage lorsque le modèle est suffisamment complet et permet une connexion au système. Ceci est le cas dans notre cadre de travail avec le logiciel ADS2 [31] qui nous a permis de valider l‟implémentation de notre contrôleur ARINC 825 et de notre terminal AFDX. La suite ADS2 a également été intégrée dans le processus de calcul et de l‟affichage de la latence des trames.

Pour pallier au manque d‟environnement de test en temps réel dans le domaine avionique, les auteurs de [32] ont développé une plateforme matérielle haute performante. Grâce au parallélisme offert, le recours aux cartes FPGA leur donne une performance de test beaucoup plus élevée que l‟approche logicielle, tout en apportant également une reconfigurabilité accrue. Leur implémentation matérielle finale repose sur l‟emploi de modules propriétaires. L‟utilisation de plateformes matérielles permet généralement de tester des systèmes en temps réel de manière directe, tandis que cette réalité doit être incluse dans tout modèle logiciel demandant un effort particulier, puisque le temps réel demeure une caractéristique importante du domaine avionique. Une mauvaise modélisation des interactions entre les systèmes en temps réel peut s‟avérer désastreuse à tout design purement logiciel. Les auteurs de [33] ont également privilégié une plateforme FPGA pour la validation d‟une nouvelle technique d‟augmentation de la vitesse de transmission à travers un bus MIL-1553. Des débits jusqu‟à 105 Mbit/s ont pu être observés, confirmant ainsi la validité de leur approche. Une preuve de concept pour le domaine avionique d‟un nouveau réseau de type « Network on Chip » a également été réalisée sur une plateforme FPGA. Leur implémentation de commutateurs AFDX [34] leur a permis de valider le respect des contraintes du domaine avionique. Cette implémentation du commutateur correspond au même type d‟implémentation que celle découlant du projet AVIO 402 [35]. Jusqu‟à maintenant, la plupart des approches matérielles tendent à ne s‟appliquer qu‟à des aspects spécifiques du

système, surtout pour en faire une preuve de concept plutôt qu'un design complet, contrairement à notre approche privilégiant la réutilisation du design, de la phase initiale jusqu'à la validation finale. En réutilisant les composants développés dans le cadre de projets antérieurs, la phase de design et de validation de nouveaux projets suivant notre approche peut s'en trouver raccourcie. Le développement initial peut donc s'avérer un peu plus exigeant lors d'une première implémentation, mais cela permet d'épargner bien des efforts une fois la plateforme matérielle configurée et les premiers composants génériques développés. Cette dernière considération peut, par contre, être abordée par l'approche logicielle bien qu'aucun de ces deux types d'approches dans le domaine avionique ne semble être privilégié ou mis en place.

2.2.3 Monitorage

Un aspect important du processus de design et de validation repose sur l'assurance du respect de requis spécifiques. Avec les méthodes de design antérieures à l'introduction du concept IMA, les performances des architectures pouvaient être obtenues aisément grâce à la modélisation ou la simulation. Avec notre approche matérielle, nous avons eu besoin d'implémenter notre propre système de monitorage pouvant répondre aux besoins spécifiques de notre projet. Dans ce contexte, nous définissons l'objectif principal d'un système de monitorage comme l'extraction des performances d'un réseau tout en réduisant au maximum son impact sur le système étudié. La présente section exposera donc les travaux pertinents dans ce domaine de recherche. Les cas plus généraux seront d'abord abordés pour venir terminer la section avec les avancements au niveau des protocoles avioniques, en particulier le protocole ARINC 825.

Bien que notre approche repose sur un support matériel pour une application avionique, l'utilisation de moniteurs est courante dans de nombreux autres domaines. Dans le domaine des systèmes embarqués logiciels sous linux, [36] introduit un outil de vérification « on-line » s'assurant de la conformité de certains paramètres d'un programme sélectionné à l'aide d'un moniteur logiciel. Développé dans le but d'être le moins intrusif possible en mode d'opération normal, ce moniteur vérifie les capteurs et les dernières données de référence et s'active en cas d'écart trop significatif. Toute mesure appropriée peut également être enclenchée à ce moment, tels un simple message d'erreur ou une réinitialisation de modules. Leur outil permet également d'effectuer les tests logiciels requis, causant malheureusement une hausse du temps d'exécution et de la consommation de mémoire. Cet outil cause donc toujours une augmentation du temps

d'exécution, bien que de manière plus réduite en mode normal d'opération. Dans le domaine du traitement d'images, les auteurs de [37] doivent assurer la surveillance de chaque image prise à haute fréquence afin de coordonner les différents signaux et de faire une meilleure corrélation temporelle. L'utilisation d'une carte de synchronisation IEEE 1588 leur permet de pallier au manque initial de leur équipement de base. Ces derniers exemples démontrent bien que les moniteurs servent à examiner les performances d'un système pouvant être utilisé pour en améliorer la fiabilité ou la qualité du processus.

Avec l'émergence de plusieurs standards de communication, tels l'AFDX et l'ARINC 825, les outils de monitorage sont devenus essentiels pour la validation des systèmes avioniques. Une revue complète des normes émergentes et traditionnelles présentée dans [38] permet une meilleure compréhension des caractéristiques à surveiller. Cette revue peut s'avérer particulièrement utile dans la sélection d'un protocole de communication dans le design d'un système avionique. Un premier moniteur d'un bus CAN est présenté dans [39]. À l'aide d'un convertisseur USB-CAN commercial, les performances simultanées de deux bus CAN peuvent être obtenues et affichées à travers un pilote (« driver ») logiciel dédié sur un PC. Le système de monitorage permet également l'insertion de stimuli afin de tester des fonctionnalités précises ou des nœuds particuliers tout en offrant un mode d'écoute complètement silencieux. Un autre système de monitorage de bus CAN est également basé sur une plateforme commerciale [40]. De manière à valider une simulation précise au bit près (« bit-accurate ») du bus CAN, les auteurs de [41] ont implémenté un réseau bus ou chaque FPGA comprend un nœud CAN. Les performances sont évaluées pour un seul des nœuds à travers un lien RS232 vers un PC permettant une surveillance sans ajouter aucun coût supplémentaire (« overhead »). Bien que la majorité des moniteurs soient conçus pour un seul standard de communication, l'outil de modélisation OPNET [42] permet la comparaison de plusieurs protocoles différents à travers la sélection de différentes configurations. Les avantages et limites de chaque standard peuvent être explorés pour le traitement de trafic spécifique. Bien qu'une connaissance approfondie des caractéristiques inhérentes à chaque standard soit particulièrement utile dans leur sélection et leur intégration, une implémentation pratique ou une modélisation logicielle est nécessaire à une intégration d'un standard pour bien s'assurer du bon synchronisme du système en temps réel et de valider la précision de la modélisation logicielle.

2.3 ARINC 825

Pour bien comprendre les contributions énoncées au niveau d'améliorations possibles au protocole ARINC 825, une courte explication présentera les principes de base d'un contrôleur de réseau ARINC 825, tel qu'implémenté dans le cadre de projet. Les derniers avancements et modifications proposées dans le domaine seront, par la suite, exposés afin de situer les contributions à ce niveau. La dernière partie viendra présenter les avantages d'utilisation d'une stratégie de partitionnement de trames, d'où l'intérêt d'étendre la norme actuelle.

2.3.1 Principes de base

La norme ARINC 825, basée sur le protocole CAN, est présentement en utilisation dans le domaine avionique. Elle offre une bonne bande passante, une gestion robuste des erreurs, une complexité relativement faible et un coût réduit. Elle n'inclut par contre aucune mesure de redondance habituellement en usage dans le domaine avionique. Ce type de réseaux permet une communication bidirectionnelle pouvant supporter la transmission de messages directement de nœud à nœud ou d'un nœud vers plusieurs points de manière simultanée. La cohérence des données transmises, d'une taille maximale de 8 octets, est assurée entre les différents nœuds garantissant qu'un message sera accepté par tous les nœuds ou aucun d'entre eux, selon les mécanismes de gestion des erreurs. Le protocole spécifie la retransmission automatique des messages erronés dès que le bus se retrouve dans un état d'attente. Chaque nœud inclut également plusieurs mécanismes de détection et de gestion des erreurs incluant la distinction entre les erreurs temporaires et les nœuds sujets à une défaillance permanente et ultimement la fermeture de nœud défectueux. Une description plus détaillée de certains de ces mécanismes est incluse dans notre 2^{e} article [10], mécanismes qui seront décrits de nouveau dans la discussion de cette thèse. À l'aide des modes d'adressage des nœuds, le réseau peut être implémenté de manière à garantir que la latence des messages prioritaires n'excèdera pas une certaine limite. L'utilisation de ce réseau procure également une flexibilité quant à la configuration, ce qui implique qu'un nœud peut être remplacé sans aucun changement au réseau, aux autres nœuds et à la couche applicative.

2.3.2 Usages

Le protocole CAN a été originalement conçu dans les années 1980 pour des applications automobiles. Par la suite, son adoption s'est poursuivie dans une majorité de systèmes de transports terrestres et commence à faire son apparition dans le domaine avionique à travers l'ARINC 825, sa nouvelle version plus fiable. L'étude des travaux sur ce protocole montre bien l'étendue d'applications de la norme de base, mais également le peu de travaux rapportés par rapport à l'ARINC 825. Même dans le domaine aérospatial, on semble prévoir une utilisation future du CAN sans aucune mention de l'ARINC 825 spécifiquement conçu à cet égard. En effet, Kimm et al. [43] vont même jusqu'à recommander l'utilisation future du CAN afin de remplacer les technologies actuelles en matière de bus en aérospatiale. Malgré cette recommandation pour l'inclusion de la version CAN de ce bus à bord de satellites, aucune littérature ne rapporte aucun exemple d'usage explicite ou d'amélioration de réseaux ARINC 825. Les travaux présentés ci-contre cibleront donc les plus récents avancements pertinents par rapport au protocole de base.

Parmi les motivations qui ont mené à des améliorations proposées au niveau du bus CAN, on peut distinguer deux objectifs principaux, soit l'augmentation de la bande passante effective disponible et le renforcement des mesures de fiabilité. Au niveau des performances, Kelkar et al. [44] ont mis de l'avant leur version de Adaptive Fault Diagnosis for Controller Area Network (AFDCAN). L'AFDCAN s'ajoute aux mécanismes existants de fiabilité en requérant l'envoi de l'état de chaque nœud à chaque cycle de diagnostic. Cette technique augmente bien la fiabilité du système au prix d'une augmentation de la bande passante requise. Afin de minimiser ce dernier impact, leur technique de réduction de données transmises leur permet d'augmenter la charge du bus de 33%, permettant la connexion d'un plus grand nombre de nœuds. L'ensemble de leur implémentation pour la validation repose sur l'emploi d'un microcontrôleur pour chaque nœud du réseau. Sheikh et al. [45] ont quant à eux proposé une approche originale en cherchant à augmenter le déterminisme des trames transférées sur le bus CAN. La cause principale de la variabilité du temps de transmission entre deux trames avec la même charge utile (« payload ») est le processus de remplissage de bits (« bit stuffing »). Le nombre de bits à insérer dépend, en effet, de la composition de la trame, soit la quantité de 0 et de 1. Afin de pouvoir mieux prédire la durée de transmission d'une trame, une section supplémentaire de taille variable vient s'ajouter à la trame CAN fixant ainsi la longueur à une taille fixe. Bien que cette mesure vienne en effet améliorer le déterminisme du protocole, qui constituait effectivement un de ces principaux points

faibles, ces nouvelles trames demandent une part plus importante de la bande passante disponible. La modification suggérée entraîne également la non-compatibilité avec la norme de base.

La méthode généralisée d'augmentation de la fiabilité d'un réseau CAN repose sur l'ajout de redondance dans le système. Les travaux présentés par [46] sur le design d'un contrôleur tolérant aux pannes sont typiques de la majorité des propositions améliorant la fiabilité de la norme. La méthode sélectionnée repose sur l'introduction locale de redondance au niveau des contrôleurs. À travers leur implémentation sur FPGA, Han et al. [46] ont simplement prouvé avec succès l'applicabilité de la redondance double dans un réseau CAN. De manière similaire reposant sur une plateforme FPGA, Szurman et al. [47] ont injecté des pannes de type SEU afin de caractériser leur contrôleur triple. Leur approche prévoit l'utilisation de redondance triple pour chaque contrôleur dont les sorties passent par un système de votation afin d'extraire les données correctes. L'applicabilité de leur méthode est ici encore démontrée bien que les gains en fiabilité due à la redondance triple sont en partie mitigés par l'augmentation significative de la taille du circuit final, résultant en une hausse de la probabilité de SEU. Une dernière approche orthogonale pour améliorer la fiabilité d'un réseau CAN prend la forme d'encryptage de données. Groza et al. [48] ont démontré l'efficacité du cryptage de données pour un réseau CAN. Leur approche a par contre l'inconvénient majeur d'ajouter du temps de traitement lors de la transmission et de la réception des trames. Ces délais additionnels réduisent donc la bande passante disponible ayant possiblement des impacts majeurs pour les applications en temps réel. Les 2 types d'approches améliorant la fiabilité d'un réseau sont donc bel et bien applicables dans le cas du protocole CAN bien que la redondance fasse augmenter de manière significative la taille finale des composants et que l'encryptage augmente la quantité de données échangées.

2.3.3 Regroupement de trames

Comme l'inclusion de stratégies de regroupement de trames est à la base d'une des améliorations à un réseau ARINC 825, cette section viendra d'abord présenter les principes de base et avantages de l'utilisation de ces stratégies. Par la suite, les différentes contributions pertinentes dans le domaine seront énoncées afin de mieux situer notre contribution à ce niveau. Un exemple simple de stratégie d'agrégation de trames dans le cadre du réseau ARINC 825 sera également présenté pour clore la section.

La plupart des systèmes distribués, comme les réseaux avioniques, sont basés sur de l'échange de données entre différents nœuds. Ces échanges sont souvent soumis à de nombreuses contraintes comme le débit de production des données, la latence maximale admissible pour chaque trame, et la taille de la charge utile. Bien que la longueur d'une trame dépende a priori du protocole de communication sélectionné, la taille des données à transmettre est généralement bien inférieure à la charge utile disponible [49]. Le regroupement de plusieurs séries de données à l'intérieur d'une seule trame constitue donc une manière de rendre le système plus efficace. La sélection de la stratégie de regroupement de trames peut rapidement devenir difficile face à la complexité grandissante des nouveaux systèmes [49]. Chaque stratégie de regroupement possède ses avantages et inconvénients. Dans leurs travaux, Kang et al. [50] proposent un nouvel algorithme de regroupement ayant pour but de minimiser la consommation de bande passante de leur réseau FlexRay [51]. De légères améliorations aux algorithmes existants, comme une meilleure utilisation du segment statique de la norme Flexray, leur ont permis d'être encore plus performants face à ces techniques antérieures [50]. La majorité des travaux pertinents ciblent une réduction de la consommation de bande passante bien que différents objectifs peuvent également être poursuivis avec le développement d'algorithmes de regroupement de trames. Le but de la stratégie proposée par [52] est de réduire la quantité d'énergie consommée plutôt qu'une réduction de la consommation de bande passante. En réduisant le nombre de bits de remplissage lors de transmissions en rafale dans un réseau WIMAX [53], les auteurs de [52] ont développé une station de base énergétiquement plus efficace. Dans un même but, Misbahuddin et al. [54] réduisent la consommation d'un réseau sans fil de capteurs. En réduisant la taille maximale de la charge utile plutôt qu'une restriction au niveau de la taille des paquets entraîne une consommation de puissance encore plus basse. Tanasa et al. [55] ont également proposé une stratégie de regroupement de trames pour un réseau FlexRay; ils ont également été les premiers à augmenter la fiabilité d'un réseau à travers leur stratégie proposée. Bien que les approches présentées [49, 52, 54, 55] ciblent particulièrement les communications sans fil et les véhicules terrestres, toute application se basant sur un réseau peut bénéficier de ces stratégies, incluant le domaine avionique. Un exemple notoire dans le domaine des réseaux AFDX utilise le regroupement de trames pour minimiser l'augmentation de la consommation de bande passante induite par un mécanisme améliorant le déterminisme du réseau [56]. En effet, bien que

l'introduction de trames tampons augmente le déterminisme du réseau AFDX, la charge sur le réseau croit également. La stratégie de regroupement vient donc réduire cet effet négatif.

L'exemple suivant prendra comme modèle un réseau de capteurs suivant le protocole ARINC 825, tel que développé dans le cadre du projet AVIO 402. Parmi les applications recensées, la stratégie la plus simple est d'envoyer une trame sur le réseau pour chaque série de données produites par chacun des capteurs. Cette stratégie n'offre aucun autre avantage que sa simplicité inhérente. Afin de réduire le surcoût, plusieurs trames ARINC 825 peuvent être regroupées de manière à augmenter l'efficacité globale. En fonction de la norme de communication utilisée, on peut réduire le surcoût en combinant plusieurs paquets. Bien que la taille des trames augmente et que la synchronisation globale puisse être influencée, la charge totale sur le réseau s'en trouve réduite. Pour illustrer les effets de ce processus, la Figure 2-5 présente 4 instances de la stratégie *M-to-1* [57] pour la transmission de 4 séries de données à travers un bus ARINC 825.

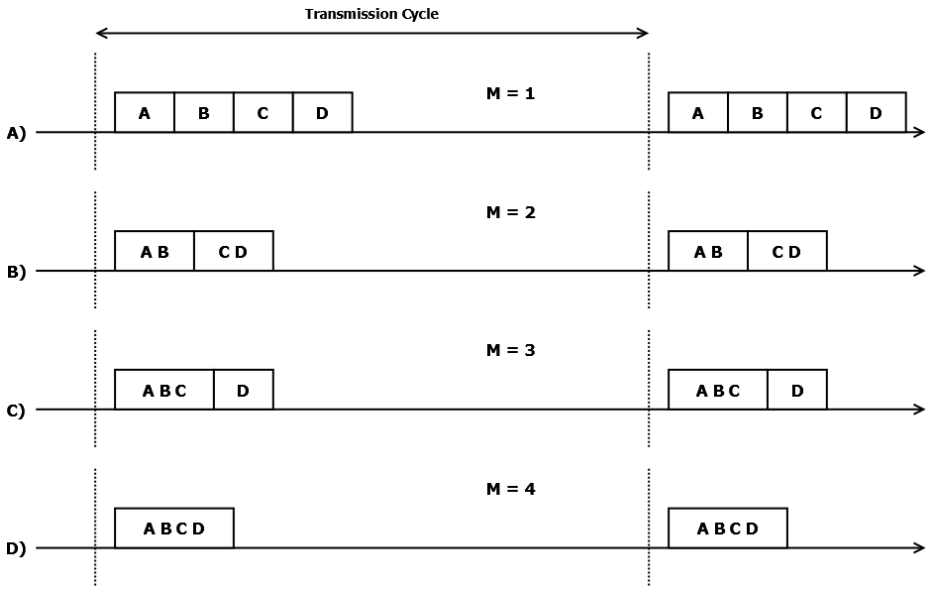


Figure 2-5: Stratégie *M-to-1*

On observe à la Figure 2-5 qu'un maximum de 4 trames individuelles peut être envoyé sur le bus pour un cycle de transmission tout en respectant la contrainte d'une charge totale du réseau inférieure à 50%. Dans ce cas (M=1), le surcoût maximal est observé pour une charge utile de 64 bits entraînant un total de 296 bits dans 4 trames séparées. On obtient un surcoût de 78% dans ce cas de figure. Lorsque M vaut 2 ou 3, la même efficacité est observée puisque 2 trames ARINC 825 sont nécessaires dans ces 2 cas réduisant quand même le surcoût à 64%. Dans le cas final

regroupant les 4 trames (M=4), le nombre de bits transmis baisse à 122 pour un surcoût final de 48 %, représentant un gain de 30% en bande passante. Cet exemple servira de base pour la partie de la discussion sur les améliorations apportées au protocole ARINC 825.

2.4 Gestion de la redondance

Dans le domaine avionique, l'introduction de redondance est à la base de la grande majorité des approches employées pour l'atteinte des cibles contraignantes de fiabilité. Le recours à la redondance demeure toujours une partie intégrante du design de systèmes avioniques malgré l'évolution vers les IMA. Plusieurs techniques alternatives seront présentées bien que leur usage dans le domaine avionique est, pour l'instant, extrêmement limité.

2.4.1 Approche classique

Traditionnellement, le niveau de redondance des systèmes avioniques varie d'un système simple sans redondance jusqu'à quadrupler le système dans son entier. Les limites pratiques, telle la multiplication des interconnexions dans des espaces réduits, viennent bien entendu restreindre l'application des niveaux de redondance élevés. En pratique, la redondance quadruple n'est utilisée que pour des applications militaires extrêmement critiques. Le Tableau 2.2 présente la correspondance approximative entre les différents niveaux de redondance et les principales cibles de fiabilité atteintes.

Tableau 2.2: Niveau de redondance et cibles de fiabilité [1]

Architecture	Non-disponibilité (Panne / Heure de vol)	Non-intégrité (Panne / Heure de vol)
Simplex	$< 10^{-3}$	$< 10^{-5}$
Duplex	$< 10^{-5}$	$< 10^{-7}$
Com-Mon	$< 10^{-7}$	$< 10^{-9}$
Triplex	$< 10^{-9}$	$< 10^{-9}$
Quadruplex	$< 10^{-9}$	$< 10^{-9}$

La non-disponibilité représente la probabilité que le système soit en perte totale malgré l'ajout de redondance, tandis que la non-intégrité indique la probabilité maximale qu'une panne ne soit pas détectée et que le système effectue son traitement habituel avec des données erronées. Les

probabilités du Tableau 2.2 sont données par heure de vol. On peut noter que l'assurance de l'intégrité du système est plus critique, puisque l'effet de continuer les opérations avec des données erronées peut s'avérer beaucoup plus dangereux que la simple détection d'une panne d'un composant. En cas de détection d'erreurs, les composants redondants sont en effet appelés à remplacer les éléments défectueux. L'architecture Com-Mon [6] implique une redondance double avec un canal de monitorage supplémentaire. L'utilisation de cette architecture est très répandue dans la communauté avionique [1]. La gestion de la redondance s'effectue traditionnellement en suivant la Figure 2-6, présentant un système reposant sur cette dernière architecture. Les autres types d'architecture sont présentés dans le 3e article [9]

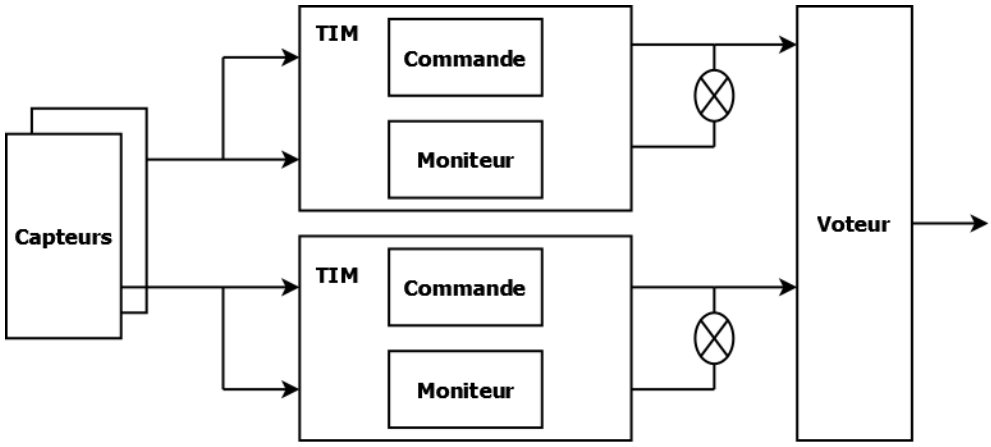


Figure 2-6: Architecture Com-Mon

Dans ce cas de figure, chaque canal possède la même légitimité avec la connexion de 2 séries indépendantes de capteurs et de contrôles. L'arbitration finale de ce type de gestion de la redondance repose donc sur un système de votation qui compare les 2 canaux. Lorsqu'un canal prend une valeur significativement différente de l'autre, la sortie du module subséquent (voteur) cesse de transmettre. L'introduction de redondance dans un système s'accompagne toujours des principes d'élimination des modes communs de défaillance. Malgré tous les efforts pour enlever toutes ces dépendances communes, la redondance entraîne nécessairement un élément commun, soit le système de votation. Dans le cas de l'architecture Com-Mon, un minimum de deux dépendances est toujours présent, puisque deux séries de comparateurs doivent être parcourues par chaque flux de données. Dans le cadre de cette thèse, l'architecture Com-Mon viendra s'ajouter à titre comparatif aux architectures et cibles de fiabilité de notre 3e article [9] dans la section discussion. Le reste de cette section se concentrera sur les derniers travaux pertinents dans

le domaine de la gestion de la redondance et quelques méthodes alternatives pour augmenter la fiabilité de tout système.

2.4.2 Approche originale

Dans le domaine des réseaux sans fil, les auteurs de [58] viennent proposer un nouvel algorithme de gestion de redondance pour un réseau sans fil. Leur algorithme permet la sélection d'un chemin redondant le plus fiable en fonction du niveau de redondance et la distance entre les nœuds. En plus d'améliorer la fiabilité de leur réseau, la consommation d'énergie s'en trouve également réduite. Afin de pouvoir utiliser un nouveau type de batterie Li-ion dans le domaine avionique, les auteurs de [59] introduisent une nouvelle approche de mitigation des effets des SEU dans le système de contrôle de leur batterie. En effet, en ajoutant un étage de filtrage, les auteurs arrivent à filtrer les erreurs dues aux SEU, bien que le filtre lui-même soit susceptible à ce type d'effets. Pour pallier à ces effets, Baronti et al. doivent tripler leur filtre et ajouter un comparateur final. Malgré tous ces ajouts, leur approche donne des résultats très semblables à une architecture de référence incluant 3 niveaux de redondance. Pour une fiabilité semblable, leur approche requiert un effort de design bien supérieur à l'introduction de redondance, mais il consomme tout de même moins de ressources matérielles, puisque la taille du filtre ajouté est de beaucoup inférieure au système de contrôle. Il est à noter qu'un point commun de défaillance sous la forme d'un comparateur est toujours présent malgré l'objectif initial d'augmenter la fiabilité du système sans avoir recours à la redondance. Plusieurs autres chercheurs ont également présenté leur proposition pour améliorer la gestion de la redondance [60-62], mais en cherchant à atteindre d'autres objectifs que l'amélioration de la fiabilité tel le raffinement de systèmes de votation réduisant le délai encouru ou diminuant la consommation de puissance. Malgré toutes les approches et améliorations présentées au niveau de l'introduction et la gestion de la redondance dans plusieurs domaines d'application distincts, le système de votation demeure un composant essentiel dont l'influence est toujours primordiale dans l'atteinte des cibles de fiabilité.

2.4.3 Approches alternatives

Puisque ces systèmes de votation viennent limiter la fiabilité des systèmes redondants, il nous apparaît important de présenter différents travaux récents pour améliorer la fiabilité des voteurs.

Une première approche pour l'augmentation de la fiabilité de systèmes de votation est présentée par [63]. En ajoutant un voteur minoritaire après chaque voteur minoritaire, leur approche permet de sélectionner le composant redondant qui ne comporte pas d'erreurs pour en propager la sortie. Sans ce voteur minoritaire, deux pannes successives dans 2 des 3 versions redondantes du même composant ne permettraient plus de déterminer le chemin encore valide. Leur stratégie a par contre comme inconvénient de consommer 2 LUT supplémentaires pour chaque voteur majoritaire. Cette approche vient faire écho à plusieurs approches énoncées dans [9] qui partage le même objectif.

Au lieu d'utiliser la redondance et de tenter d'améliorer la fiabilité du voteur, plusieurs ont plutôt choisi des méthodes alternatives à la redondance pour mitiger l'effet des SEU. Une première méthode, appelée « scrubbing », consiste à profiter de la reconfigurabilité dynamique des FPGA pour contrer l'effet des SEU. En reconfigurant le FPGA et en comparant la nouvelle configuration avec l'ancienne, la présence d'erreurs peut donc être détectée. La nouvelle approche de « scrubbing » développé par [64] induit un gain en fiabilité comparable à la technique de base tout en consommant 16 fois moins de mémoire. Les auteurs de [65] proposent une revue complète les différentes stratégies possibles de mitigation des SEU. Une première stratégie pour mitiger les SEU repose sur une réduction directe de la menace. Les techniques de « Hardening » [66] sont un bon exemple pouvant réduire jusqu'à 3 fois l'influence des SEU. L'introduction d'une large couche P+ à forte concentration sous la région active des transistors [67] donne des résultats similaires en induisant par contre un délai supplémentaire. La deuxième stratégie consiste à éviter les conditions d'opérations rendant les équipements vulnérables. Par exemple, la sélection de certains types de FPGA (Flash, Anti-Fuse) qui peuvent s'avérer plus résistants face au SEU, mais plus susceptibles à d'autres formes d'erreurs. Finalement, le recours aux infrastructures peut s'avérer efficace en implémentant des procédures dans des systèmes où l'on prédit des erreurs. Les protocoles de gestion des erreurs de l'ARINC 825, tel l'encodage CRC, sont un bon exemple de ce type de procédures en permettant la détection d'erreurs et la retransmission des trames erronées. Malgré l'inefficacité de la redondance au niveau des voteurs dans une optique de réduction des points communs de défaillances, il existe quand même plusieurs autres approches pour en augmenter la fiabilité. Ces techniques requièrent souvent un effort de design supplémentaire par rapport à la simplicité inhérente de la redondance.

Les deux prochains chapitres présenteront chacun un article présentant les contributions réclamées dans cette thèse. Le chapitre 5 viendra présenter les discussions relatives aux articles contenus dans les chapitres 3 et 4. Pour chacun des axes de contribution concernés, un bref des sommaires des articles sera d"abords présenté. La conclusion viendra par la suite compléter la thèse.

CHAPITRE 3 ARTICLE 1 : A SYSTEM ARCHITECTURE FOR SMART SENSORS INTEGRATION IN AVIONICS APPLICATIONS

José-Philippe Tremblay, Yvon Savaria and Guchuan Zhu

École Polytechnique de Montréal

Claude Thibeault and Safwen Bouanen

École de Technologie Supérieure

Publication : SAE International Journal of Aerospace, Volume 5, No 1, pp 187, Octobre 2012

Abstact : With the next generation of avionics systems, more sensors and actuators will be required for an ever increasing number of functions. In this paper, we propose a system architecture based on several enhancements to the IEEE 1451 standard, granting it a wider application range, improved resource efficiency and a generic and reusable character. This architecture facilitates the integration of next generation smart sensors with a wide range of avionics data communication networks and allows the specification of generic features to be respected. In order to meet the requirements of avionics applications, this architecture that provides a design framework offers customization of features such as bandwidth, reliability, resources utilization and compatibility with different types of transducers, especially smart sensors. The resulting resource utilization and reliability are analyzed for several configurations that provide a basis for comparison. To validate the proposed architecture and the benefits it offers, we have designed and implemented a transducer network inspired by representative avionic needs. The implementation reported in this paper targets a LX45T Xilinx FPGA board. The transducers are connected to the data network through field buses based on the ARINC 825 protocol, while the backbone of the network is based on the AFDX specification. The analysis of the ensuing prototype shows an important increase in reliability that result from using the proposed architecture. We also show that this architecture enables important complexity reduction over a typical transducer network based on the same communication protocols for the same level of reliability.

3.1 Introduction

With the next generation of avionics systems, more sensors and actuators will be required for an ever increasing number of functions. Future avionics platforms will also need to manage communications from different application domains, inducing an important increase of the information flow throughout an aircraft [68]. Moreover, stringent safety and reliability requirements are also complicating the situation, by demanding at least the duplication of most on-board avionic systems. Based on the current state-of-the-art in the avionics field, the design of a suitable network meeting all constraints would be very time and effort consuming [69]. The ever increasing diversity in the transducer market makes it even harder to meet the requirements, since most of them are tailored for some specific class of applications. Although some control network and field bus architectures are available to interface these transducers, they still lack the capability to easily support different types of transducers. The process of integrating transducers to these networks requires significant effort and is thus very costly. These considerations set a requirement for a more effective and systematic approach to design transducers and their associated network interfaces.

The IEEE 1451 standard has been developed several years ago to propose a reference model for the control of smart transducers [7]. Although this promising standard provides guidelines to facilitate the integration of a smart transducer to any type of network interface, analysis has shown that some key features were missing for avionics application. In particular, concerns about setting and respecting reliability targets are the main considerations lacking with the basic application of the standard. This is the main motivation for this paper in which we propose a system architecture based on several enhancements to the IEEE 1451 standard granting it a wider application range, improved resource efficiency and a generic and reusable character. This architecture facilitates the integration of next generation smart sensors with a wide range of avionics data communication networks and allows the specification of generic features to be respected. In order to meet the requirements of avionics applications, this architecture offers customization of features such as bandwidth, reliability. The proposed design framework allows optimizing resources and offers compatibility with different types of transducers, especially smart sensors.

The rest of the paper is organized as follows. The “Related Work” section first presents the basic concept of the IEEE 1451 for further reference throughout this paper. The advantages and limitations of the architecture are highlighted, which forms the comparison basis for the proposed architecture. This section also reviews some variations on the basic IEEE 1451 introduced by other researchers to provide points of reference. The “System Architecture” section presents the proposed architecture and the improvement over the typical IEEE 1451 architecture. Reliability issues are addressed in the “Reliability” section. The first part of this section recalls the method used to predict the reliability of each of the elements composing the network. Based on these last predictions, the second part of this section presents the guidelines to obtain reliability estimates for the considered network architecture. The results obtained based on a prototype implementation, as well as observations and analyses, are presented in the “Results‟ section. Finally, concluding remarks and discussion on the improvement offered by the proposed solution are provided in the “Conclusions” section.

3.2 Related Works

As mentioned before, the proposed architecture for the integration of transducers for avionics applications is an extension to the IEEE 1451 standard. The suite of IEEE 1451 smart transducer interface standards was developed several years ago to address issues related to the need for plug-and-play smart transducers. Nowadays, sensors and actuators are ubiquitous in many engineering and scientific systems. Since transducers are used in so many different scenarios, their design is often tailored to a specific class of applications. The integration of transducers within a network based on a current field bus technology may require significant efforts and may lead to different advantages or weaknesses depending on the characteristics of transducers and of the system where they are used. Supporting various types of networks and protocols or producing multiple transducer versions for every relevant network specification on the market would require significant efforts and cost from a manufacturer. This can be mitigated through universally accepted sets of open standards.

The goal of the IEEE 1451 standard is to allow access and control of smart transducers through normalized interfaces. The use of microprocessors or any other dedicated electronic hardware to handle digital communication has also opened the opportunity for adding intelligence to sensors and actuators. According to IEEE 1451, in order to be qualified as a smart transducer, the device

needs to fulfill three specific features. The transducer itself must be defined by a machine-readable transducer electronic data sheet (TEDS). The TEDS is a nonvolatile memory, located in the transducer, used for storing transducer identification, calibration, correction data, measurement range, and manufacture-related information. The second feature states that control and data associated with the transducer must be in a digital format. Finally, triggering, status, and control must be provided in accordance with the TEDS‟s information to support the proper operation. The type of transducer, either actuator or sensor, does not affect its qualification as a smart transducer. Nowadays, smart sensors are slowly starting to make their way in some applications, mostly in less restrictive fields. They are currently been considered in avionics, but have not yet been adopted [70]. The proposed interfaces allow transducers to be connected to a processor, an instrumentation system or any other control network. The reference model, shown in Figure 3-1, is completely independent from communication protocols and hence, it is also applicable to wireless networks. We note that this reference model is very similar to structures found in avionic transducer networks with the exception of the interoperability inherent to the use of the standard.

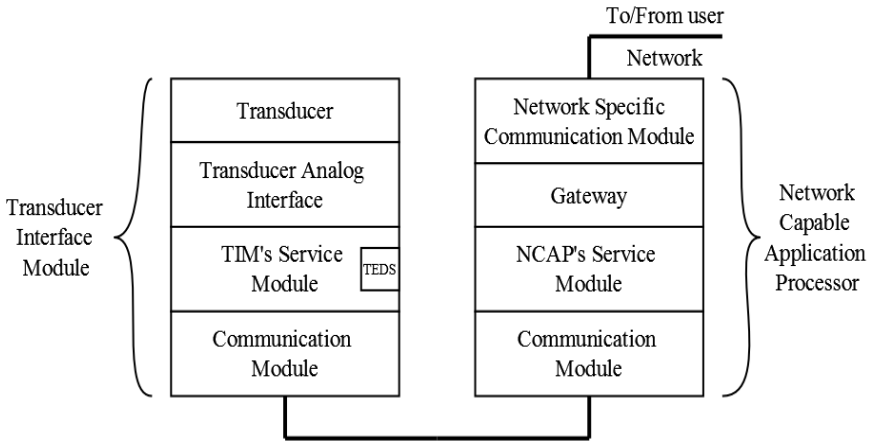


Figure 3-1: IEEE 1451 Reference Model, adapted from [7]

The functionality proposed in the standard is separated into two main modules. The transducer interface module (TIM) has the following responsibilities: analog signal conditioning, digital conversion, command processing, data transfer, and TEDS storage. The transducers, from a single sensor or actuator to many heterogonous units, are enclosed within the TIM. The second part of this model is the network capable application processor (NCAP) linking the TIM to user networks. The NCAP is responsible for interface control, command routing, data correction, and

message coding and decoding. The physical interface between TIM and NCAP is left to the designer‟s choice and is not covered by the standard. The third part of the standard (IEEE 1451.3) is applicable for the definition of an interface for a multi-drop system using distributed communication architecture, regardless of the selected field bus technology. This scheme globally allows a NCAP to offer services to several TIMs sharing the same communication link. A simplified version of this rather specific reference model for a distributed architecture is presented in Figure 3-2.

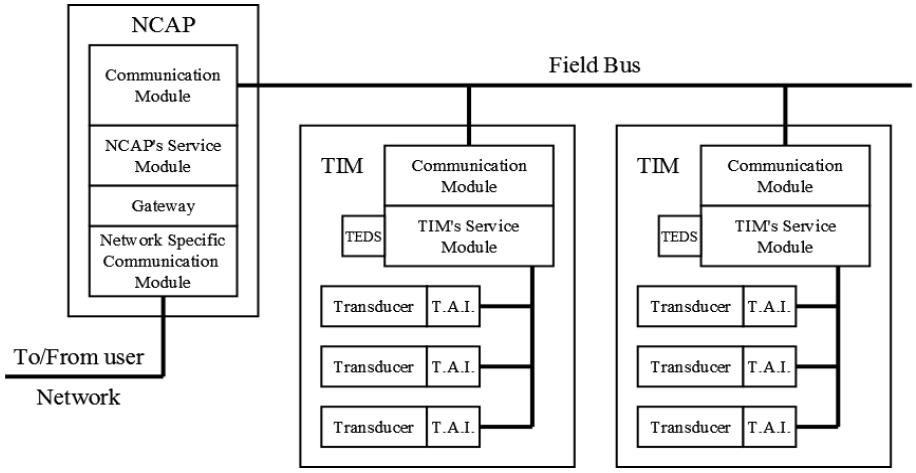


Figure 3-2: IEEE 1451 Distributed Architecture

The adoption of this standard provides several advantages in the design of transducer networks. The main benefit is to ensure compatibility between any combination of transducers, network protocols, and manufacturers. Having a reference model and a framework to follow for the control of transducers and the data management enables another important gain in productivity by reducing the design effort and the development cost. The multi-drop portion of the standard also guarantees a plug and play capability making it easier to install, update, replace, and move any component in the system. For avionics applications, the use of data buses and communication networks can eliminate wiring burdens as the current on-board architecture relies on a multitude of point to point connections. However, our preliminary research has shown that the basic framework provided by the IEEE 1451 standard lacks the ability to reach the reliability requirements for the stringent aerospace environment. This drawback will be demonstrated in the “Results” section. Although one can readily benefit from the generic aspect, improving the reliability, as we propose in this paper, is essential for the adoption of this standard for avionics applications.

Although IEEE 1451 offers clear advantage, several researchers have also found weaknesses inherent to the standard and proposed their own improvements. A dual connection based on a CAN bus is proposed in [71] to improve the reliability of an automotive system. Although this approach is not based on duplication of all key modules, it does increase the reliability of the system. However, that improvement is not sufficient for fulfilling all requirements of aerospace applications. Their proposal is also limited by the specified number of field bus while our proposal makes our architecture completely generic. A method for autonomous self diagnosis in an IEEE 1451 based architecture has also been proposed [72]. Their approach is based on the introduction of a fault detection layer within the NCAP structure. The effect of this improvement on the network‟s complexity has yet to be analyzed. Other than reliability, energy consumption has been one of the main important research topic related to IEEE 1451. An improved method for grouping transducer to reduce the power consumption of the network is proposed in [73]. The principles formulated in this last paper were proven for wireless communication, but they are also applicable to a wide range of wired implementation. These works also demonstrate that even though the IEEE 1451 standard provides a generic architecture to ease design and integration, several aspects limit its application to some specific fields, such as avionics systems.

3.3 System Architecture

Figure 3-3 presents a global view of a version of the proposed architecture. This architecture has been designed to be as generic as possible. In this architecture, our NCAP and TIM modules have the same functionality as the one described in the basic IEEE 1451 specification.. The number of field busses is proportional to the number of communication module included in each TIM and NCAP. The number of NCAP is selected in order to reach the required level of reliability. The addition of more NCAPs only results in an increase in redundancy, while the bandwidth remains unaffected since the amount of exchanged data remains unaffected. The number of TIMs is mainly dictated by the number of transducers and their location. Several TIMs are served by one or several NCAPs, as needed to fulfill the reliability requirement. This number of TIMs served by a NCAP is limited by the available bandwidth of the combined field busses.

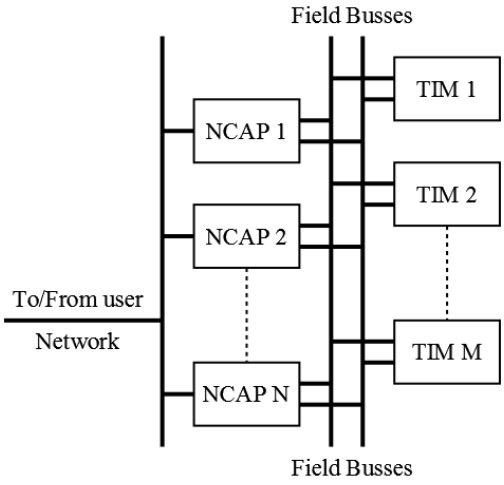


Figure 3-3: Custom global architecture

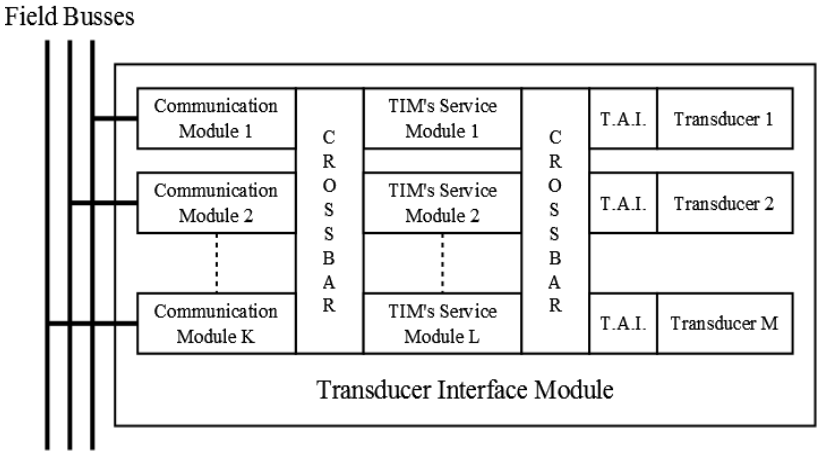


Figure 3-4: Custom TIM‟s architecture

The proposed internal structure of the TIM, presented in Figure 3-4, can support numerous configurations of transducers by changing the number of its internal components (K, L and M). The TIM is composed of parallel configurations of similar functional modules separated by crossbars. The role of these crossbars is to provide a data path from each transducer to any functional module. In the case of a module‟s failure, processing can then be done by the other copies of the same type of module. This eliminates many single points of failure. Although the redundant field buses are primarily introduced to increase the system reliability, they also increase available bandwidth. The number of service and communication modules is thus a function of the reliability requirements and of the minimum needed bandwidth. Each transducer is connected to the crossbars through its own transducer analog interface. The NCAP‟s structure, shown in Figure 3-5, is similar to the TIM.

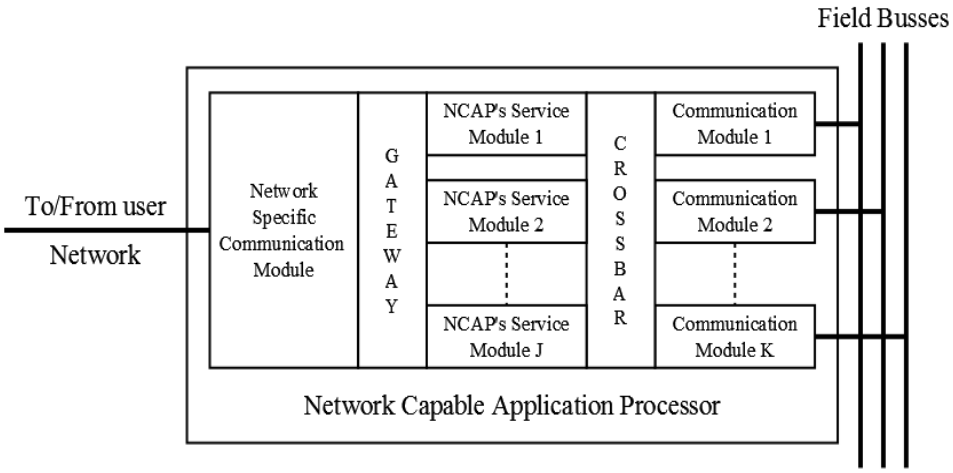


Figure 3-5: Custom NCAP"s architecture

3.4 Reliability Evaluation

Before estimating the reliability of the various considered configurations, some background concepts and terminology are reviewed. In the present work, the mean time between failures (MTBF) has been adopted as the measure of reliability. The MTBF is a commonly used concept to estimate an electronic component"s reliability, especially in avionics applications. In order to evaluate the MTBF of the various proposed and considered architectures, we relied on a common modeling methodology. The modeling process starts by the construction of a hierarchical diagram for the 3 architectures discussed earlier, namely basic, distributed and custom (see Figure 3-1, Figure 3-2 and Figure 3-3). The modeling of the basic and distributed architectures was done solely for comparison purposes. These diagrams were then converted into mathematical equations representing the reliability relation based on the configuration of their basics constituents. The 2 main configurations, serial (eq. 3.1) and parallel (eq. 3.2), are expressed by the following mathematical statements [74].

$$R_S = R_1 \times R_2 \times R_3 \times \cdots \times R_N = \prod_{i=1}^{N} R_N \qquad (3.1)$$

$$Q_S = Q_1 \times Q_2 \times Q_3 \times \cdots \times Q_N = \prod_{i=1}^{N} Q_N \qquad (3.2)$$

$$R = 1 - Q \qquad (3.3)$$

where R_n represents the probability of success or reliability of the n^{th} item and Q_n corresponds to its unreliability or probability of failure of each component. R_s and Q_s stand for the system as a whole. Based on the assumption of a constant failure rate, the unreliability is then equal to the

MTBF of the selected element. The relationship between R and Q is expressed in (3.3) the reliability and unreliability of a basic module. To complete the reliability estimation of the proposed architecture, the last equation needed corresponds to the case where *k* out of *n* items of equal reliability, organized in a parallel configuration, must be functional in order for the global system to be operational. The reliability for this structure is derived from the above equations applied to the proposed configurations. The k of n parallel topology that contributes to improve the reliability is present in the 3 different levels of our architecture: the service modules level, the communication modules level and, at the NCAP level itself. For service and communication modules, the minimum number of functioning element (k) is regulated by bandwidth requirements imposed by the number of transducers. For example, if a single service module has sufficient bandwidth to serve 6 transducers on a sustained basis, at least 2 operational service modules would be required within a TIM containing 9 transducers. In the case of the NCAP, only one module is necessary, since any increase in the bandwidth requirements is already handled by the duplication of communication modules. In other words, the insertion of more than 1 NCAP is used exclusively for reliability purposes rather than for improving the bandwidth of the system.

Finally, the last missing information for a complete reliability analysis is the MTBF prediction for each of the elements in the architecture. For this task, we used the military handbook for reliability prediction of electronic equipment (MIL-HDBK-217) [75]. Within the avionics and military fields, this handbook is the most widely known document for reliability prediction. It contains failure rates specification for numerous electronic components such as integrated circuits, transistors, relays and many more. The obtained predictions take into account several parameters such as complexity, packaging, environment and quality. The reliability predictions reported in the “Results” section have then been computed using analytic expressions derived using the guidelines of this handbook. The derivation and full details of the reliability equations will be reported elsewhere.

3.5 Results

To validate the proposed architecture and to measure its benefits, we have designed and implemented a hardware based prototype of a transducer network. The implementation targets a LX45T Xilinx FPGA. Through the use of several of these platforms, we validated our approach by implementing of the TIM and the NCAP described in the previous sections. Although the

implementation is hardware based, the principle guiding the design of the architecture can also be applied to a software based design. We have selected the ARINC 825 protocol to implement the field bus as it allows multi-drop bi-directional communications. The resulting prototype is configured to maintain a throughput of a least 1 Mbit/s and to guarantee a deterministic traffic over the field bus. The user network not covered in this paper follows the AFDX specification.

The first step of the analysis process is the complexity evaluation and reliability prediction of the basic components composing the architecture. Tableau 3.1 presents the characteristics of the implemented modules. The communication module is an ARINC 825 bus controller. The service module covers the most important parts of the IEEE 1451, such as data formatting, command structure, and TEDS access. The bus interface, only present in the distributed TIMs, follows the I2C interface as it is one of the most commonly used protocols for this low level communication. To minimize the influence of the crossbar to the overall system reliability, its internal structure has been duplicated. This added redundancy has a low impact on the total complexity of our proposed architecture while preventing the crossbar from being a bottleneck in terms of reliability.

Tableau 3.1: Modules complexity and failure rate

Module	LUTs	Register	Failure Rate
Communication	1164	1578	3.815e-07
Service	345	379	3.741e-07
Bus Interface	230	379	3.731e-07
Crossbar	32 * Nb_port	0	1.619e-13

Based on our derived analytic reliability models of the different architectures and the individual failure rate, we have estimated the reliability of the basic and distributed architectures to, respectively, 1.51×10^{-6} and 2.26×10^{-6} failures per flight hour for each transducer. As the data path does not change in accordance with the number of transducers, these estimations remain constant for a network of arbitrary size. We can also see that the inherent simplicity of the basic architecture makes it more reliable than its counterpart. Note that the communication bus is a single point of failure in the distributed TIM, which is often forbidden in aerospace systems. For comparison purposes, we have tailored our architecture‟s reliability to match the respective

reliability estimates of both references for each number of transducers. Note that in the conventional base architecture, each additional transducer requires an additional TIM, which is not the case with the distributed architecture and with our proposed architecture. The relations between the complexities of the respective networks of same reliability for each given number of transducers are presented in Figure 3-6 and Figure 3-7. Since the results are fairly similar for the variation in the amount of required registers, the figures focus specifically on the number of LUTs.

As shown in Figure 3-6, our custom architecture consumes significantly less resources than the basic application of the IEEE 1451 standard to meet the same reliability target. This is mainly explained by the inherent reutilization of computational resources in the proposed architecture compared to the full replication of components for each transducer. The benefits grow with the number of transducers. For a subnet composed of 24 transducers, as shown in Figure 3-6, our proposed solution offers a 87% reduction of the complexity for the specific case of our implementation. This observation is noteworthy as most of the current avionics networks rely on a point to point connection for each transducer. The ARINC429 protocol is a good example of this connection scheme for wide spread currently used airborne technologies.

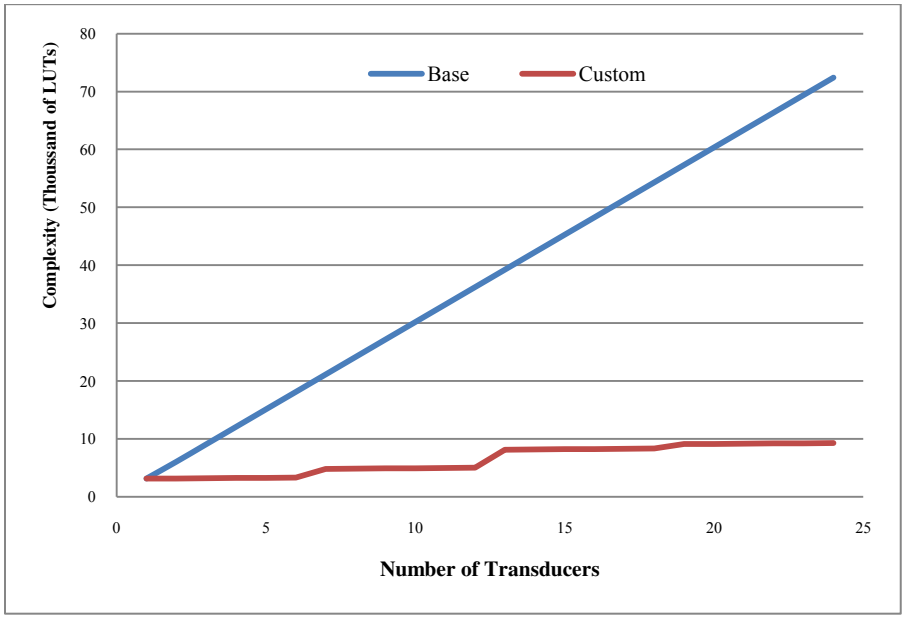


Figure 3-6: Network Complexity Vs. Number of Transducers

In the case of the distributed architecture presented in Figure 3-7, we always manage to match the network‟s complexity for the same level of reliability. Our approach allows reaching lower

failure rate targets by simple means of configuration. A practical limitation we faced in our experiments came from the limited number of IOs available on the FPGA board, as a common FPGA can easily contain the network for large numbers of transducers. Increasing the number of transducers in a TIM enables greater reduction but requires additional IOs for connection to the FPGA. Several bigger FPGA devices providing the necessary number of IOs are currently available. For the connection of 24 transducers within the same hardware framework, the LX150T with a larger FPGA would be sufficient. A bus interface, such as the I2C, could also solve this problem, but this would also lower the reliability as shown with the distributed architecture.

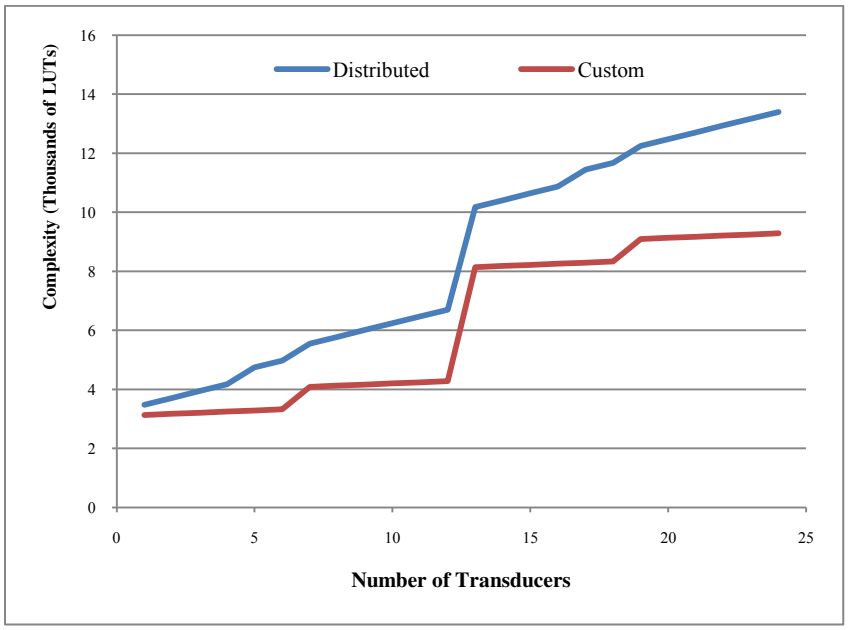


Figure 3-7: Network Complexity Vs. Number of Transducers

Within the context of our research project, we had to follow a requirement stating that the electronic communication network had to offer a failure rate better than 10^{-6} failure per flight hour. Figure 3-8 shows the necessary complexity in order to fulfill this last reliability requirement. Based on Figure 3-8, our architecture is again more effective than both the point to point connection and the distributed architecture. Duplication has been used to reach this reliability target for the basic and distributed architectures. The main drawback of this traditional technique is the doubling of the network‟s complexity, as well as the number of required transducers. Our architecture would then require the installation, maintenance and repair of only

half of the transducers of the distributed network. We consider this to be significant in the design of an avionics transducer network in order to reduce its size, cost and overall complexity.

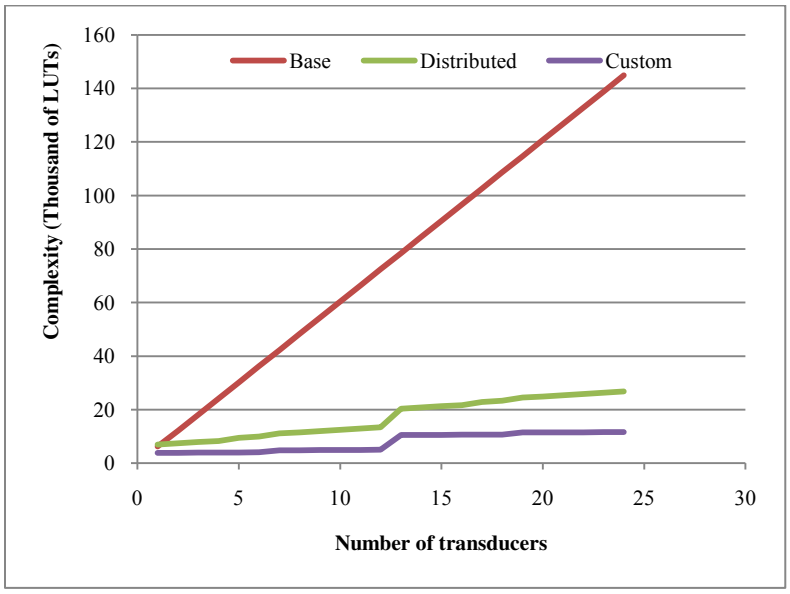


Figure 3-8: Network Complexity Vs Number of Transducers

3.6 Conclusion

In this paper, we have proposed a system architecture based on several enhancements to the IEEE 1451 standard granting the design a wider application range, improved resource efficiency through a generic and reusable framework. Our approach enables to tailor the network‟s architecture to reach the desired levels of reliability through efficient resource consumption. Compared to traditional designs, our proposed architecture can help reduce the complexity by 87% compared to the basic IEEE 1451 architecture for a 24 transducers network for the same level of reliability. This gain is even more important when trying to reach a more stringent requirement. The inclusion of a generic reliability feature in the proposed architecture allows meeting this requirement without having to rely on the duplication of the whole system. In the future, this research will explore how to optimize the matching process of our architecture‟s reliability to a specific target. This process is currently done manually but its automation would help guaranteeing the lowest resource consumption for the same reliability requirement.

CHAPITRE 4 ARTICLE 2 : HIERARCHICAL REDUNDANCY MANAGEMENT FOR AVIONIC NETWORKS

José-Philippe Tremblay and Yvon Savaria

École Polytechnique de Montréal

Claude Thibeault

École de Technologie Supérieure

Publication : Aerospace and Electronic Systems Magazine, IEEE, 2016

4.1 Abstract

With the adoption of the Integrated Modular Architecture for avionic systems that must comply with stringent reliability requirements, redundancy management has become a crucial aspect of the design of any avionic network. This paper presents an approach to integrate redundancy into a network based on the utilization of a hierarchical architecture allowing the appropriate inclusion of redundancy at any specified layer. Benefits and limitations of this architecture are presented in comparison to the most traditional redundancy management schemes. As demonstrated, the hierarchical approach explored in this paper allows reaching the required reliability targets, while being more efficient in terms of resource utilization. Experimental results obtained with a hardware platform confirm the validity and the applicability of the presented approach.

4.2 Introduction

The adoption of the IMA approach [76], currently used in most of the recent avionic systems, has brought many improvements upon a simpler but limited federated architecture. The introduction of shared resources through high bandwidth data networks has enabled a massive reduction in wiring throughout aircrafts. However, this massive reduction is also associated with an added connectivity between systems, which is a main concern with the IMA approach. Indeed, stronger connectivity could also allow error propagation on multiple systems or network malfunction affecting simultaneously several systems. By comparison, with its dedicated communication lines, the federated architecture inherently offered independence between systems. The natural tendency for any system engineer is to approach the IMA architecture through proven methodologies, to avoid mistakes and to take advantage of lessons learned [76]. This introduces

significant inertia in areas such as redundancy management [77]. It is thus important to understand where new projects differ from past projects and require adjustments to improve design practices.

Through our work that focused on the design of a transducer network [10] based on a new architectural interpretation of the current IMA concepts, we have identified some aspects of the design process that we believe could be improved over previous approaches. One such important aspect of avionic system design is the need to provide reliability assurances. Up to now, reliability targets are typically achieved with the simple duplication of every individual components of the system [74]. It has become the most common means for achieving high reliability targets with complex electronic systems, such as avionics. In this paper, we hence present our approach to redundancy management that helps determine the most appropriate amount of redundancy for each specific network layer. While optimizing the network‟s architecture, we are able to reduce complexity by grouping compatible hardware structures and eliminating single points of failure. The approach investigated in this paper also allows validating the impact of different means of introducing redundancy in various protocols. The design of a transducer network composed of a main AFDX network, supported with several ARINC 825 field busses, will serve as a reference point to illustrate the principles and benefits of our approach.

The rest of the paper is organized as follows. A related work section first presents some research related to the integration and management of redundancy. It will mainly present the traditional approach to introduce redundancy in avionic systems based on the IMA and federated approaches. Then, section III introduces our approach to redundancy management. This approach will be demonstrated through the design of a typical transducer network. Our previously reported FPGA-based implementation of this network [8] [10] is then used to illustrate the application of our proposed approach. Based on this implementation, realistic gains in resource efficiency will be characterized in the implementation results. Finally, concluding remarks and discussion on the benefits offered by our approach, notably in terms of performance improvements, are provided in the last section.

4.3 Avionics Networks Redundancy

4.3.1 Redundancy Management

The management of redundancy within any type of system has been an active area of research in the past years, leading to several advances related to the work presented in this paper. It has always been an essential part of the design of all avionic systems targeting critical applications. Depending on the specific requirements and nature of the application, several means of introducing redundancy can be adopted to increase the reliability of a system. Each redundancy model comes with its own advantages and limitations, and its integration within a complex system can sometimes be an arduous task. In order to manage faults in a system, we can rely on strategies based on either spatial or temporal redundancy. Time redundancy executes a task several times in order to compare the results through a voting system [78]. With time redundancy, single event upsets can be detected and corrected through the redundancy introduced by repetitions. In terms of reliability, this redundancy management strategy also implies that both the processing element and the voting system become single points of failure making them inadequate for direct use in avionics in their basic form. Although significant advances have been presented in [79] to reduce the sensitivity to the reliability of the voting system and processing element, the added latency incurred for the repetition of each task greatly limits the bandwidth required for real-time applications. Due to these limitations, the spatial or modular redundancy, based on the modular repetition through a dedicated hardware system, have been privileged for avionic systems. The parallelism granted by hardware redundancy can mitigate the increased load induced by time redundancy. Although some parts of the hardware could still be considered single points of failure, our paper will show that through the use of a suitable hierarchical architecture, high reliability targets can be met for avionic applications. Since the reliability of voting systems used in modular redundancy becomes critical, several authors have presented means to mitigate this sensitivity [62, 78, 80, 81]. These last works, from different domains of application and implementation methods, allow the selection between multiple reliability targets with trade-offs in design complexity. The system impact of the voting system required for the inclusion of any type of redundancy will be covered in more details in the sections to come.

Our review focuses next on mainstream applications of spatial redundancy management schemes, followed by achievements specific to avionic applications. Comprehensive reviews of traditional

spatial redundancy management schemes are presented in [82] [83]. These reviews are useful to address the issue of identifying the advantages and drawbacks of each redundancy model in order to deliver the required functionalities. The differences between active and standby approaches as well as the effects of modular duplication are well discussed in [82], while others [10, 46, 47] provide a more practical approach to redundancy validation. With the FPGA development of a Triple Modular Redundant (TMR) CAN controller, the authors in [47] presented the effects of single-event upsets (SEUs) within their designs. Their fault injection process is based on the direct manipulation of the FPGA bitstream to emulate the appearance of SEUs. Their experiment showed that for a large number of injected faults, a TMR controller became less reliable than the original single version. A larger occupied area on the FPGA and the introduction of a single point of failure (the voting system) make the TMR implementation less reliable than a simplified theoretical analysis predicts. Two distinct implementations [10] [46] of a dual CAN controller and their validation have also been presented to improve the reliability of a single controller. While both implementations satisfy their own real-time requirements with high reliability, they also differ in their approach (Active/Standby) and the quality of service in terms of data loss and synchronization. These various approaches to redundancy management were shown to lead to acceptable designs, while providing different benefits and drawbacks. Although the presented works [10, 46, 47] mostly focus on the duplication of the CAN controller, these approaches can be used in several application domains relying on data exchange over a network.

In the avionics domain, several redundancy management schemes have also been presented. With their implementation of an interface between two space vehicles, McCabe and al. [84] presented their analysis of necessary considerations required for the implementation of an IMA architecture for critical applications. In order to guarantee the reliability of their interface, they have duplicated functional and communication elements through a typical lockstep mechanism. By bridging two vehicles relying on different design approaches (IMA and Federated), they showed [84] that different avionics architectures with different redundancy management schemes can be interfaced in a fault-tolerant, reliable manner. The work of Forsberg [85] and Haouati [86], both targeting the design of transducer networks, proposed two successful validation processes of architectures devised through different approaches. The approach presented in [85] starts by designing a non-redundant functional architecture and then adds redundancy in a later stage of the process. At this step, each component is duplicated and included in the final validation

through theoretical reliability analysis. To design their network, the authors of [86] rely on the strict methodology put in place by Airbus to respect the reliability requirements. Every processing element is systematically and asymmetrically doubled to ensure fault detection. These papers are good examples of commonly used practices for the design of complex avionic systems relying on basic redundancy of each component to meet reliability targets.

In a traditional avionic design flow, such as the one presented in [85], redundancy is directly integrated in the architecture in the same manner for each module. Upon selection of the redundancy management strategy, it is usually applied without any distinction at every layer, with a final theoretical validation. The two main recognized, studied and utilised form of redundancy are component and system redundancy. Component redundancy is based on the local replication of modules, while the implementation of system redundancy rests on the complete replication of the system. Briefly, the parallel structure of the added modules typically reduces the failure probability, while additional serial layers increase the failure probability. For a parallel structure S, where K out of N components must be functional, the probability of success (assuming a perfect voting system) is given by equation 4.1 [74], where R is the reliability of each parallel branch.

$$R_S = \sum_{i=K}^{N} \frac{N!}{i!(N-i)!} R^i (1-R)^{N-i} \qquad (4.1)$$

In fact, it is proven that component redundancy provides a greater reliability than system redundancy in most situations [87]. It is to be noted that, in all cases, be it with component or system redundancy, a reliable voting method is required for redundancy to be used successfully. As was shown in the related works, a voting system with the suitable reliability can be selected using the appropriate design approach. In avionics, the presence of a voting system becomes a coupling factor between otherwise independent modules or systems. Unlike some fields of application where this factor does not matter so much, its presence can cause a drop in effectiveness of some otherwise valid redundancy management scheme. The redundancy management scheme presented later has been devised to mitigate the coupling effect. The effect of the coupling on the system‟s reliability and the effectiveness of the presented approach will be discussed in section IV.

4.3.2 Federated and Integrated Modular Architectures

In order to better understand our approach to redundancy management, a short presentation of the basic principles behind the Federated and IMA architectures will be covered in this section. The outdated federated approach to the design and development of avionic systems, adopted with the introduction of airborne electronic components, relies on the independent implementation of dedicated functions onto self-contained hardware platforms. These individual units are then connected through simple point-to-point connections for exchange of data and coordination signals. In avionics, the ARINC 429 field bus standard [2] is commonly used in this type of architecture. Independent unidirectional communications for each component of the network, granting an inherent determinism, is a main feature of a federated architecture. An example of this architecture is depicted in Figure 4-1 with the typical connection of 2 sets of sensors to a Flight Control Computer (FCC).

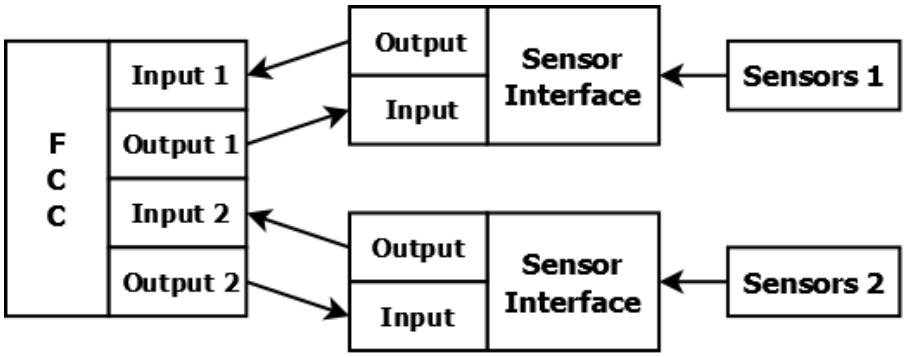


Figure 4-1: Federated Architecture

The federated architecture somehow eases the integration process, since only a simple verification of the available bandwidth and timing constraints is required. The inherent separation between functions and platforms is also greatly advantageous in meeting the stringent reliability constraints of aircraft systems. Although the simplicity of the architecture offers several benefits, its limitations started to appear over 2 decades ago. The first major drawback comes from the increased complexity of on-board airplane functions.

Figure 4-2 presents the Airbus A320 flight deck display system, structured a typical federated architecture requiring different levels of redundancy throughout the same architecture. It is an example of typical airliners before the integration of the IMA concept. Its architecture exploits multiple ARINC 429 busses. In Figure 4-2, triple redundancy is required for the main systems

(DMC, EFIS), while double redundancy is required for the other avionic components. Connection with non-redundant non-avionic systems is also present. Although several levels of redundancy are required for each system depending on their critical nature, the mismatch of redundancy levels does not incur any change to the design and validation aspects since all systems are still independent and their failure cannot affect directly any other systems.

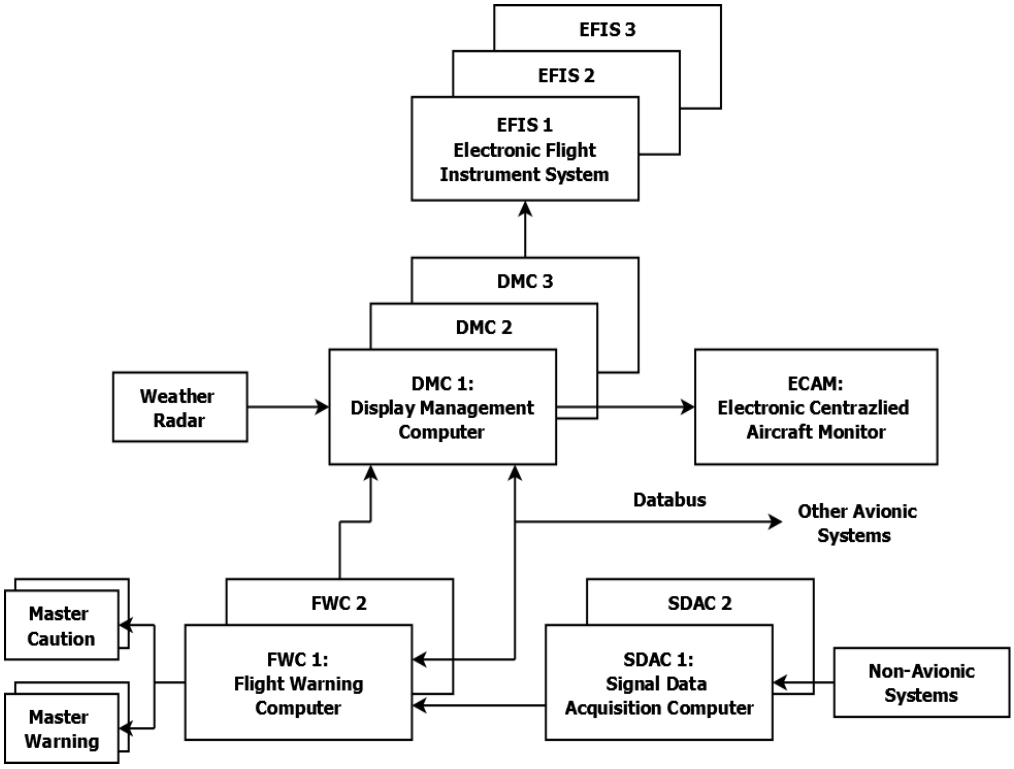


Figure 4-2: Airbus A320 flight deck display system [1]

Nowadays, functions are increasing in complexity and can sometimes be distributed over several systems or even be dependent on each other. For example, this is the case with fly-by-wire already used in many aircrafts that can provide a completely electronic control of an airplane. Such systems require tight interactions between multiple functions and dedicated units. With federated architectures, the multiplication of individual components and their associated communication lines resulted in large costs and bulky wiring. These are the main reasons for the industry shift to the IMA architecture that relies on the distribution of functions over several hardware components sharing a set of communication channels. A possible generic representation of an architecture offering services similar to the federated architecture of Figure 4-1, but implemented with the IMA approach, is presented in Figure 4-3.

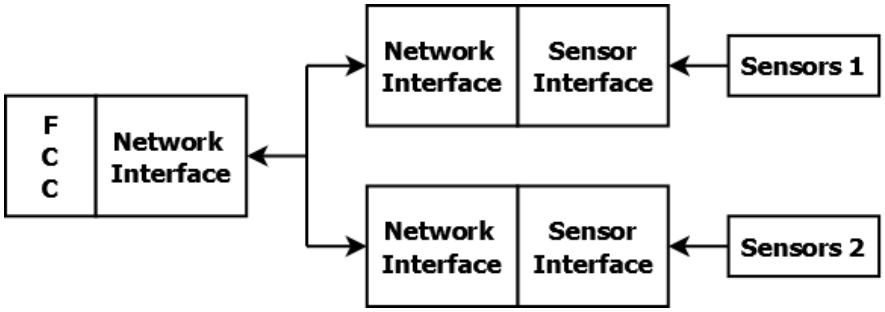


Figure 4-3: IMA Architecture

As we can see by comparing Figure 4-1 and Figure 4-3, the IMA architecture appears to reduce the necessary resources as there are fewer modules and physical busses. Additional sensor clusters could easily be added to further leverage the shared network connection. A cluster represents a group of any number of sensors sharing the same initial connection to the network. Sensors within a cluster are usually located close to each other. This opens possibilities to minimize the amount of resources consumed by sharing components and selecting communication standards providing enhanced performances, such as the AFDX protocol [3] that is gaining in popularity. This migration towards the IMA approach requires an evolution of methods and tools in order to maximize its benefits. The next section will present our approach to redundancy management developed in the context of the optimization of the tradeoff between resources and reliability. In comparison to traditional redundancy management approaches, such as the ones presented in the beginning of this section; our approach has been specifically developed for use in conjunction with the IMA architecture.

4.4 Redundancy Management

Due to the insufficient reliability provided by each of the individual network constituents, we investigated approaches that would allow integrating redundancy into our implementation of an avionic sensor network. Our network is based on the local connection of the sensors to a field bus, following the ARINC 825 standard. Sensor data is then encapsulated in frames by gateways before being forwarded onto a main AFDX network towards the FCCs. More information on the overall requirements and hardware implementation of the network analyzed and characterized here can be found in [10]. The work reported here is partly motivated by a first attempt at introducing redundancy by following traditional approaches, namely system and component

redundancy. These approaches were found to be inefficient for the proper integration of two distinct communication standards, ARINC 825 for the field buses and AFDX for the core data network. Basic redundancy is already present within an AFDX network while the ARINC 825 standard has mechanisms to check data integrity but does not include any form of redundancy that can deal with failed buses or bus interface. This motivated us to evaluate at each layer of the network the proper way to integrate redundancy through a proposed generic architecture [8]. It was found that this redundancy management method can take advantage of the inherent capabilities of each standard, as demonstrated in the rest of this paper. The functional architecture on which our redundancy management method is applied is first presented to serve as a reference for comparing the traditional system and component redundancy management schemes with our proposed approach. These two traditional schemes and our approach will then be applied to the functional architecture to identify their advantages and limitations. In this context, it is to be noted that each of these three resulting redundant architectures nominally require the same effective bandwidth, since they serve the same number of sensors. Nevertheless, the actual bandwidth requirement is dependent on the required redundancy level.

Within the context of our project, our sensor network relies mainly on the integration of an ARINC 825 as the field bus on the periphery of an AFDX main network [10]. Of course, several field buses could be used on the periphery of the AFDX core serving as a backbone network for data and control information. As envisioned, field busses would connect together selected sensors and actuators, while the high speed AFDX network would relay all relevant information from/to the flight computers. A non-redundant version of the functional architecture for a single cluster of the network is shown in Figure 4-4A.

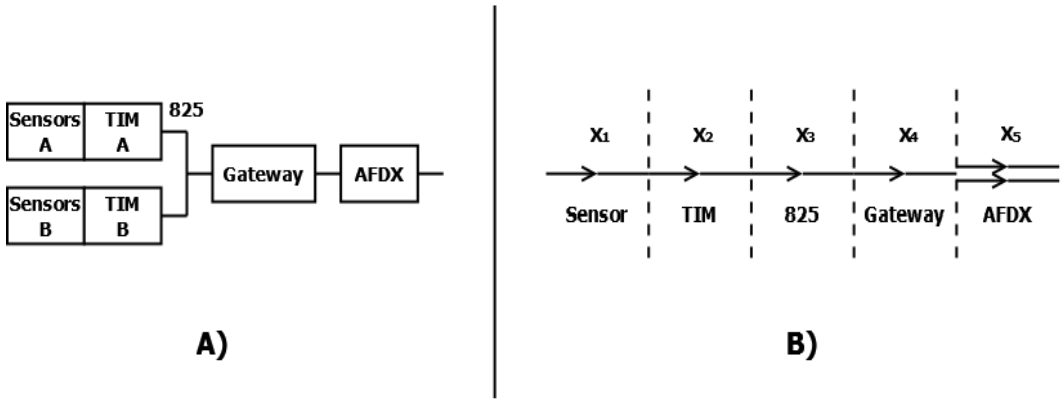


Figure 4-4: A) Functional architecture, non-redundant version B) Associated reliability model

The Transducer Interface Module (TIM) is mostly responsible for data acquisition and actuator control. These two tasks are performed by parallel processing elements (not shown). Based on the selected network architecture [8], the number of parallel processing elements within a TIM can vary according to the number of sensors that need to be connected, the throughput required and the reliability targets. At the ARINC 825 layer, we considered the controller to be included within the 825 layer, although it is physically included in the TIM or the gateway. The gateway itself is in charge of message routing and linking of the two relevant frame formats, AFDX and ARINC 825. The ARINC 825 field bus is configured to maintain a throughput of 1 Mbit/s and can accommodate up to 12 sensors. The AFDX network can sustain a bandwidth of 100 Mbit/s, which is sufficient for the connection of several clusters onto the main network depending on the network topology. More details on our implementation of the network can be found in [10].

The reliability model related to the (functional) non-redundant architecture is shown in Figure 4-4B. Each of the components responsible for a part of the transmission of the frame then becomes a layer, denoted X_n, in the reliability modeling of the studied architectures. The overall reliability of the architecture can be obtained by evaluating the reliability of each individual layer and computing the system‟s reliability through existing reliability theory [74, 87]. The reliability R of any individual component represents the probability of a correct computation and transmission through the associated item. The reliability analysis takes into account permanent faults as well as the first transient fault not being handled by existing fault management mechanisms, such as the CRC encoding used in the ARINC 825 protocol. The source of the permanent faults can either come from regular usage (bus disconnection, normal breakdown) or from a transient fault having permanent effects such as SEU in a configuration table causing all subsequent frames to be sent to an incorrect destination. Any fault considered would either cause a module to produce some faulty results or to be completely disabled. The reliability of the basic constituents has been computed with equation 4.2, linking the failure rate of a module λ_i and its reliability R_i.

$$R_i = e^{-\lambda_i} \qquad (4.2)$$

The individual failure rate of each module has been obtained through typical avionic estimates [75] incorporating known causes of failures for each type of component and implementation.

Based on the model of Figure 4-4, we can extract equation 4.3 to represent the reliability of the non-redundant architecture taking into account the reliability of all layers.

$$R_{\text{Non-redundant}} = R_{\text{Sensor}}\, R_{\text{TIM}}\, R_{825}\, R_{\text{Gateway}}\, R_{\text{AFDX}} \qquad (4.3)$$

The reliability of the overall non-redundant architecture, defined as the probability of the FCC to receive the correct data produced by the sensors, simply corresponds to the product of all individual reliabilities. For comparison and to demonstrate the effectiveness of the presented redundancy management scheme, the reliability of the studied architecture will be computed for two distinct objectives: error detection or data correction. For the detection objective, the reliability is defined as the probability to receive the correct data produced by the sensors or to receive an incorrect data flagged as erroneous by the voting system. In the case of data correction, only the reception of the correct value is considered as a successful transmission. The main difference between those two reliability objectives lies in the number of functional paths required. Indeed only one correct data path is required for error detection, while having a majority of the paths functional is required for data correction.

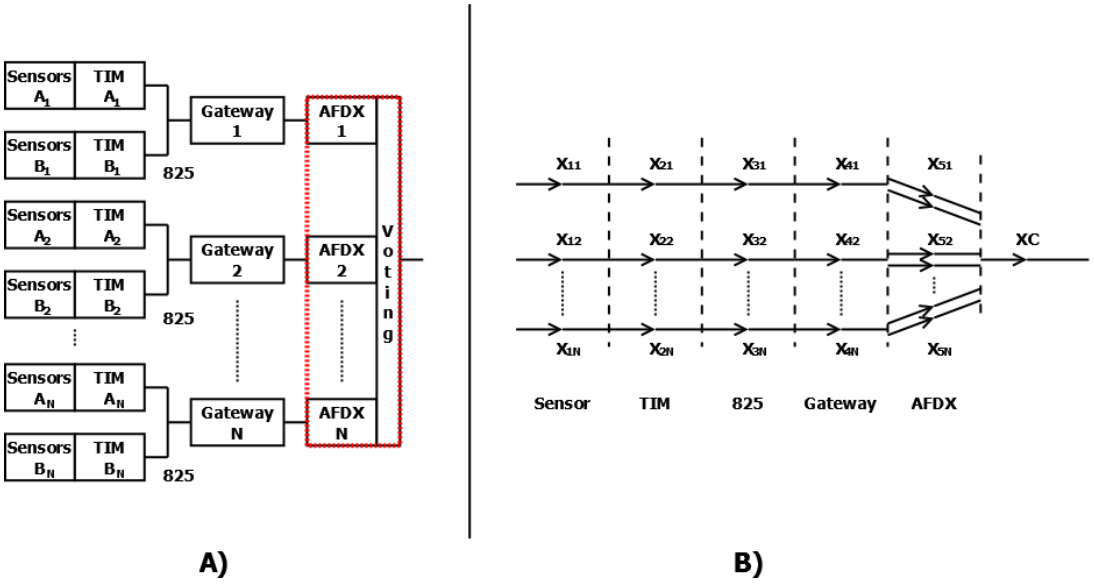


Figure 4-5: A) System redundant architecture, B) Associated reliability model

In order to satisfy reliability targets, several redundancy management schemes have been applied as part of this research on the functional (non-redundant) architecture presented in Figure 4-4 to better understand the advantages and limitations of each scheme for application in avionics systems. A first reliable version of the architecture designed based on a complete system

redundancy and its associated model are presented in Figure 4-5. The rest of the figures in this section focus on the presentation of generic architectures in terms of the number of spare modules, while components enclosed in red rectangles are considered to be physically coupled together.

In the system redundancy scheme (Figure 4-5A), the complete system is replicated N times. For each required level of redundancy, another totally independent network is added. The independent networks are then coupled at the end with an extra layer. This layer called the coupling, denoted XC, links the networks that are otherwise independent. This coupling can be accomplished with a suitable voting system [80, 81, 83]. The coupling in an avionic system is governed by the need, if possible, to eliminate all single points of failure. In this regard, the selection and application of the system redundancy scheme has the advantage of concentrating areas where single point failures are a threat. Based on equation 4.1, equation 4.4 represents the reliability associated to the system architecture of Figure 4-5.

$$R_{System} = R_C * \sum_{i=K}^{N} \frac{N!}{i!(N-i)!} R_{NR}^{i} (1 - R_{NR})^{N-i} \qquad (4.4)$$

where R_C is the reliability of the chosen coupling element, R_{NR} is the reliability of the basic non redundant architecture, N is the level of redundancy (namely the total number of paths) and K is the minimum number of functional paths required based on the chosen reliability objective. For error detection, at least one path is required, while a majority of functional paths are required for data correction. Figure 4-6 presents another type of redundancy management called component redundancy.

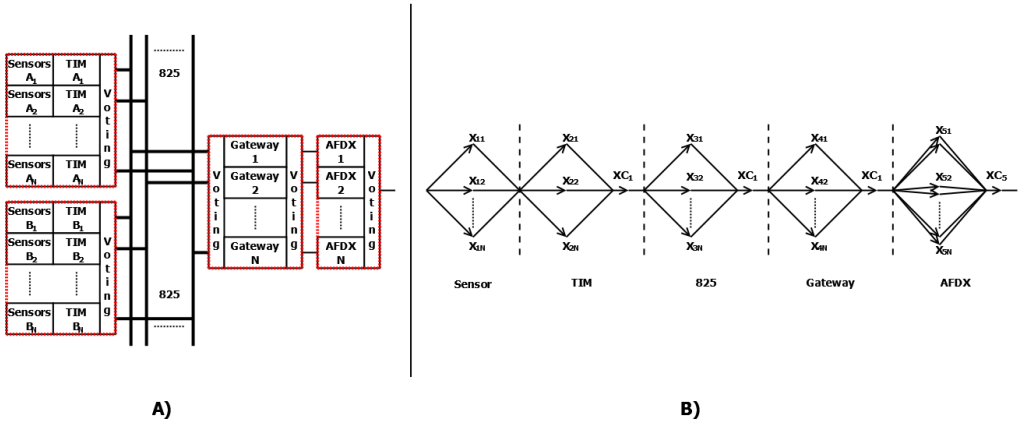


Figure 4-6: A) Component redundant architecture B) Associated reliability model

Although local modular replication is supposed to be the best management scheme in most situations, we can see the appearance in Figure 4-6 of a coupling at each layer. These couplings represent the added effect of the voting system and the common factor between each component and its redundant version(s). Unlike the architecture with system redundancy, these common factors are due to sharing of some hardware components that can introduce dependencies. Note that, in practice, features such as power supply sharing can introduce significant dependencies and that can be considered in a coupling term. Equations 4.5 and 4.6 are used to compute the reliability of the component architecture from Figure 4-6.

$$R_{Component} = \prod_{L=1}^{5} R_L \qquad (4.5)$$

$$R_L = R_{CL} * \sum_{i=K}^{N} \frac{N!}{i!(N-i)!} R_{BL}^{i} (1 - R_{BL})^{N-i} \qquad (4.6)$$

where R_L is the reliability of the layer L, R_{CL} (still) the reliability of the coupling element for layer L (the values of this parameter can of course change depending on specifics of an architecture), R_{BL} is the reliability of the base component of layer L, N is the number of available components or paths and K the minimum number of required functional paths for either detection or correction. The reliability of the component architecture represents the probability that each layer is reliable. In this case, a layer is considered reliable if its coupling element is not faulty and if at least K basic components are not faulty.

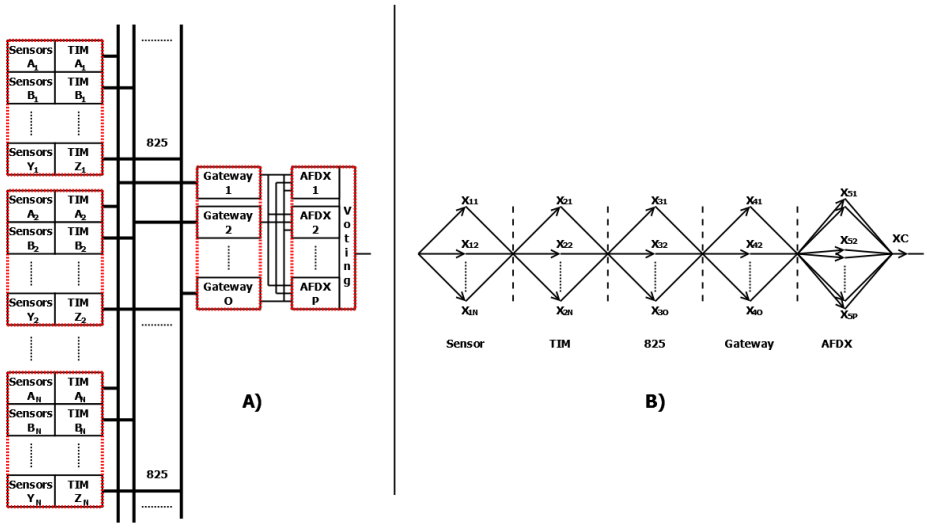


Figure 4-7: A) Hierarchically redundant architecture, B) Associated reliability model

To complete our analysis, the architecture of our implemented hierarchical network and its associated model are provided in Figure 4-7. In Figure 4-7 A), each red box encloses components sharing dependencies, such as power line and implementation on the same FPGA. We also define a cluster in our hierarchical architecture as the regrouping of the TIMs and its connected sensors like the leftmost red boxes from Figure 4-7 A).

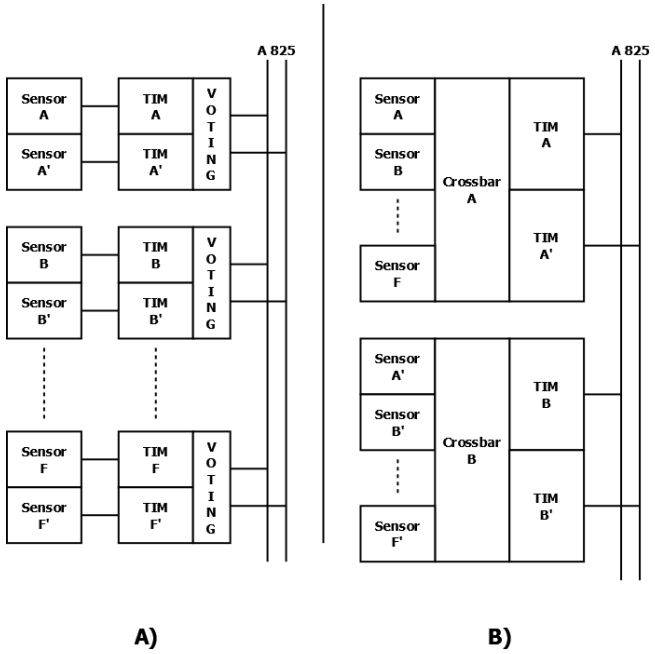


Figure 4-8: TIM Structure A) Component Architecture B) Hierarchical Architecture

In order to benefit from the hierarchical architecture, careful grouping of the sensors within the TIMs must be devised. In Figure 4-8, a possible grouping of 6 sensors and TIMs for the component and hierarchical architectures is presented to better understand the complexity analysis of section IV. We see that each pair of sensors is connected to a pair of TIMs and a voting system for the component architecture. By making sure that a sensor and its redundant counterpart are not within the same physical cluster, the hierarchical architecture allows eliminating all dependencies between components. Indeed, all acquired data and their redundant version(s) are now passing through completely independent components. Within a cluster of the hierarchical architecture, it is to be noted that the TIM modules can be shared since the data flows from each sensor within a cluster are independent from each other. This allows reducing the number of TIMs per cluster, as a single TIM is able to serve several sensors within its allocated

processing time, unlike previous architectures, where the need to avoid possible single point failures imposed dedicating an individual TIM for each sensor.

At the gateway layer, any common dependencies must be carefully eliminated. In this case, gateways should be implemented on separate platforms with independent power supplies as well. At the AFDX layer, additional redundant components are not necessary since an AFDX network already includes redundancy, as denoted by the double arrow in this layer in Figure 4-7 B). Indeed, in AFDX networks, messages are sent as two redundant frames over the network in order to compensate for any loss of frames due to some fault (including timing and soft faults for instance). To counter a permanent switch and link malfunction, AFDX requires individual End Systems to communicate over multiple independent and redundant networks. As we can see in the End System specification from Figure 4-9 [3], each End System receives and eliminates redundant frames from two independent networks (A and B).

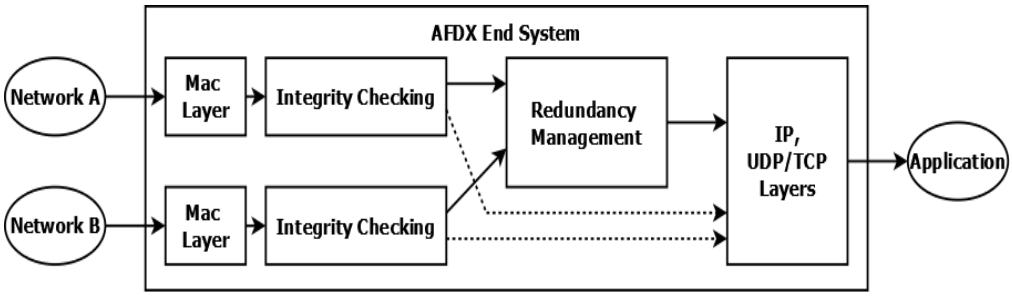


Figure 4-9: AFDX End System Specification

This characteristic of the AFDX network partly explains why the connection to a single AFDX switch may already be sufficient to ensure the required reliability. Through the use of the hierarchical architecture, it is then possible to modify the level of redundancy at each layer instead of considering the system as a whole as in the system and component traditional architecture. As will be shown in the following sections, the proposed approach requires fewer resources compared to both traditional (component and system redundancy) approaches. Equations 4.7 and 4.8 are used to obtain the reliability of the hierarchical architecture.

$$R_{Hierarchial} = R_{CL} * \prod_{L=1}^{5} R_L \qquad (4.7)$$

$$R_L = \sum_{i=K_L}^{N_L} \frac{N_L!}{i!(N_L - i)!} R_{BL}^i (1 - R_{BL})^{N_L - i} \quad (4.8)$$

Where, as in Eq. 4.6, R_{CL} is the reliability of layer L, but here R_{CL} is the reliability of the coupling element implemented as part of the AFDX layer, R_{BL} is the reliability of the base component of layer L, N_L is the redundancy level of layer L and K_L the minimum number of required functional paths of layer L for either detection or correction. The main difference with the component architecture modeled by Eq. 4.6 is that only the AFDX layer includes the coupling‟s reliability to represent the final voting (or redundancy management) system. All the other couplings have been removed through the use of the hierarchical approach, as we can see in Figure 4-7A). It is to be noted that, unlike with traditional approaches, the redundancy level N_L of each layer as well as the minimum number of required functional paths K_L can vary depending on the requirements. The following section will provide some details on the validation platform used to obtain the results allowing the proper analysis on the benefits of the proposed redundancy management scheme.

4.5 Hierarchical Architecture Analysis

4.5.1 Reliability estimations

Based on the architectures and associated models, reliability estimates for each redundancy management scheme have been computed and compared. The failure rate of each independent component was obtained in relation to our FPGA-based hardware implementation of the network [10]. The sensor interfaces, the ARINC 825 controllers and the AFDX gateway were implemented on 2 SP605 boards from Xilinx, while the AFDX end system was designed by enhancing a regular Ethernet protocol stack with the functionalities specific to AFDX. The validity of the implementations of both standards has been confirmed with the ADS2 software [31]. ADS2 is a powerful modular family of software components, designed to support the development, prototyping, test, and validation of avionics communication networks. Communication has been successfully established with ADS2 through the ARINC 825 field bus and AFDX network to validate our implementation of our ARINC 825 controller and AFDX end system. The only missing failure rate was for the AFDX layer composed of commercial switches. For this layer, we selected a value well below the typical failure rate of the other components to

reflect the redundant nature of the AFDX network. For the first part of our analysis, we selected a value of $4*10^{-9}$ as the failure rate of the AFDX switches. We also chose a value of $4*10^{-7}$ for failure rate of the coupling elements, representing a worst case scenario for the same type of implementation as the rest of our implementation. The coupling element includes all the forms of dependencies that are present, such as the voting system, the power supply and any other. For simplicity in the following numerical evaluations, all couplings are assumed to have the same reliability. In a first step, the failure rate of the AFDX and coupling elements have been fixed, while the second part will analyze the sensitivity of the architectures to their variation. Based on the redundancy management scheme presented in Figure 4-5, Figure 4-6 and Figure 4-7, our analysis includes both the error detection and data correction management schemes. We set our analysis to 3 levels of redundancy (N, N_L) to enable the analysis of the data correction management scheme. Tableau 4.1 presents the failure rates obtained for each presented redundancy management scheme. The failure rate for the architecture without redundancy of Figure 4-4 has been computed to $1.204*10^{-6}$, which is better than the reliability obtained with the component redundancy approach, but worse than the reliability calculated for the system and hierarchical approaches. As a redundancy level of 2 is insufficient for data correction, the associated failure rate is not applicable (N/A) in Tableau 4.1.

Tableau 4.1: Architecture's failure rates

Architecture	Related Figure	Failure Rate	
		Error Detection	Data Correction
Component (N=2)	5.6	1.60000 E -06	N/A
Component (N=3)		1.60000 E -06	1.60000 E -06
System (N=2)	5.5	4.00001 E -07	N/A
System (N=3)		4.00000 E -07	4.00004 E -07
Hierarchical (N_L=2, L=1 to 4) (N_5 = 1)	5.7	4.04000 E -07	N/A
Hierarchical (N_L=3, L=1 to 4) (N_5 = 2)		4.00001 E -07	4.00002 E -07

Although the hierarchical architecture provides reliability similar to that calculated for the system architecture, the hierarchical approach enables a gain in resource efficiency, as it will be shown in Tableau 4.2. As we can see from, the gain in reliability for error detection between dual or triple modular redundancy is very limited for an increase of about 50% in resources consumption. Although triple modular redundancy would seem like a more reliable redundancy management scheme, the presence of the coupling factor dominates the overall network's reliability. In this case, the main benefit of triple redundancy then resides in its data correction capability. In all cases of data correction, triple redundancy offers reliability values similar to error detection, due again to the impact of the coupling mechanisms and associated terms. In terms of architecture, we see that both system and hierarchical architectures provide the best reliability as their related failure probabilities are practically equal. Because of the inclusion of the coupling effect at each layer, the component architecture becomes the least reliable. The architecture without any added redundancy even provides a better reliability than the component architecture. We can conclude that the component redundancy approach may not be suited for avionic systems, even though this approach to redundancy has been proven to be superior in most situations [74].

4.5.2 Complexity analysis

Tableau 4.2 presents the amount of resources consumed in the implementation of each traditional network for a case study of 12 connected sensors implementing standard dual redundancy. As different methods of implementation can have significant impact on the required resources, we focused our complexity comparison on the required number of individual modules. Efficiency can then be examined layer by layer. In the case of the traditional inflexible architectures (component, system), one TIM is required for each sensor. For these cases, we can easily see that both the component and system architectures consume twice as much resource as the non-redundant architecture, which was expected because of the choice of dual redundancy.

Tableau 4.2: Number of components for the traditional architectures

	No redundancy	Component	System
NB TIM	12	24	24
NB 825 buses	1	2	2
NB Gateways	1	2	2
NB AFDXs	1	2	2
Voting Systems	0	4	1

The results on resources consumption for the hierarchical architecture are presented in Tableau 4.3. Due to the hierarchical nature and elimination of physical dependencies of the presented approach, several sensors can share a single TIM through the appropriate separation between each sensor and its redundant counterpart. Our implementation of the TIM service module can process data coming from up to 4 sensors within its allocated processing time. Therefore, 1 TIM module is required for every 4 sensors connected to the cluster. The only drawback of sharing the service module is the need for a simple crossbar to enable a connection between each sensor and service modules. In terms of complexity, the crossbar is significantly smaller than the service module. Although added crossbars are required for the implementation of the hierarchical architecture, the reduction in required resources due to a smaller number of TIM‟s modules is far

more than the added resources required for the implementation of the crossbar. In terms of our reliability analysis, the failure probability of the crossbar is directly included in the worst case reliability of an individual TIM module, as presented in the beginning of section IV.

Tableau 4.3: Number of components for the hierarchical architecture

Component	Hierarchical Redundancy		
Sensor	4	8	12
TIM	2	4	6
825 bus	2	2	2
Gateway	2	2	2
AFDX	1	1	1
Crossbar	0	1	2
Voting System	1	1	1

Considering a system with 12 sensors implemented according to the hierarchical architecture, we observe from Tableau 4.3 a gain in resource efficiency at the TIM and AFDX layers when compared to traditional dual redundant architectures. In terms of voting system, both the hierarchical and system architectures rely on a single voting system, while the component architecture requires a voting system per layer, increasing the amount of resources for a lower overall reliability. At the AFDX layer, a duplication of the AFDX module was not necessary for the hierarchical architecture, due to its redundant nature that could be taken into account during the design process. Due to the granularity of the redundancy for each layer of the hierarchical architecture, a complete AFDX network can then be eliminated, thus saving significant design efforts and costs.

For the TIM layer, proper grouping of non-redundant sensors allows redundant modules to be used for increasing the reliability, as well as the available processing power. As we can see by comparing Tableau 4.2 and Tableau 4.3, for the same number of sensors, the hierarchical

architecture requires 4 times fewer TIMs than traditional redundancy management schemes for a set of 12 sensors. As the hierarchical architecture provides reliability similar to the system architecture, the hierarchical approach is clearly more effective in terms of resources. The analysis of the hierarchical architecture presented in Tableau 4.3 for different numbers of sensors highlights another benefit of the approach. As we can see, the number of components remains the same for different number of sensors. With respect to the number of TIMs for instance, connecting 9 or 12 sensors would require the same number of components. An ARINC 825 bus supporting 9 sensors could then easily be upgraded to 12 sensors, according to the maximum number of sensors allowed, without significant modifications, unlike the case with traditional architectures. The hierarchical architecture then allows more graceful expansion or modification capabilities.

4.5.3 Coupling's impact on reliability

The following analysis characterizes the sensitivity of the network to the inherent reliability of the ADFX layer and the coupling effects. Figure 4-10 and Figure 4-11 present the unreliability (failure rate) of each architecture under a variation of the unreliability of the AFDX component. The reliability of the architectures in Figure 4-10 has been computed to implement an error detection management scheme while data correction is the focus of Figure 4-11. Globally, a reduction in the failure rate of the AFDX layer causes the network"s reliability to improve for all studied cases up to a certain point, where any further improvement of the basic AFDX component would be ineffective. The regions where improvements of the AFDX does not improve system reliability is present in both figures. It corresponds to the plateaus observed at low AFDX unreliability values. The plateaus are mainly due to the layer with the lowest reliability that tends to dominate the computed reliability. In this case, the coupling elements become the components with the lowest reliability since every other system component"s reliability improves through redundancy. In Figure 4-10, the plateaus for dual redundant architectures begin around a failure rate of 10^{-7} for the AFDX component, which is slightly smaller than the coupling unreliability that is in the order of 10^{-6}. For triple redundancy, these plateaus seen in figure 10 and 11 appear around a failure rate of about 10^{-3}. At the point of operation, for failure rate of $4*10^{-9}$ of the AFDX component, both the hierarchical and system architecture provide the best reliability as we have seen in the analysis of Table 4.1. At this same

operation point, the presence of multiple coupling elements explains the low reliability of the component architecture that becomes even worse than the non-redundant architecture. When decreasing the reliability of the AFDX component, we can observe a greater sensitivity for the hierarchical architecture than with the other architectures with active redundancy. The impact of this sensitivity is lower for triple redundancy than for double redundancy. The impact of the degradation in the AFDX components becomes more important for the hierarchical architecture than with the traditional redundancy management schemes. The tailoring allowed by the hierarchical architecture enables more efficient use of the resources at the expense of greater sensitivity to the reliability of layers with lower redundancy. It is to be noted that, in Figure 4-10 the obtained reliability values for the dual system and triple hierarchical architectures are too close to be distinguishable.

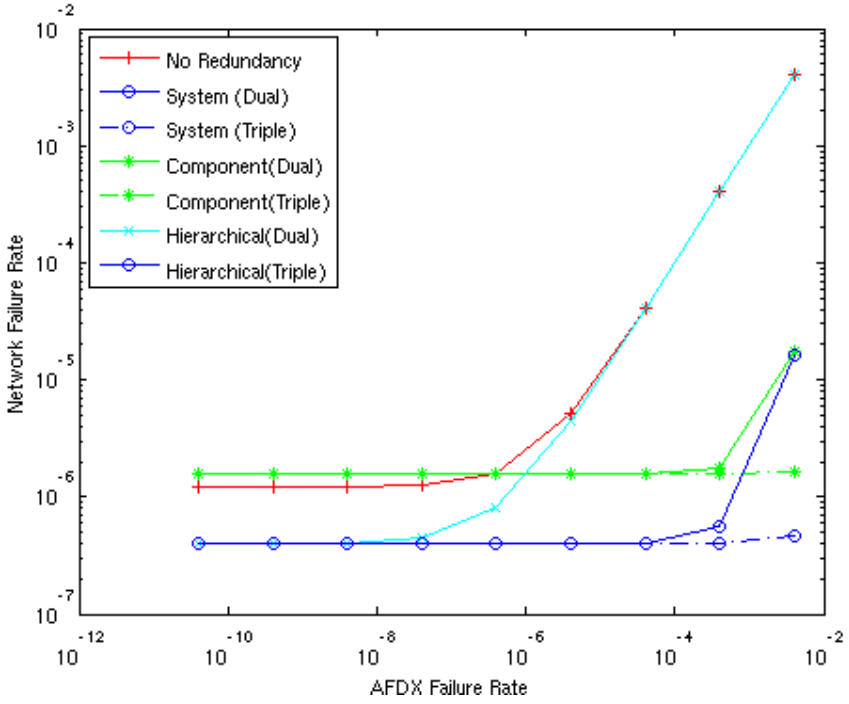


Figure 4-10: Architecture‟s unreliability as a function of the AFDX‟s unreliability for error detection

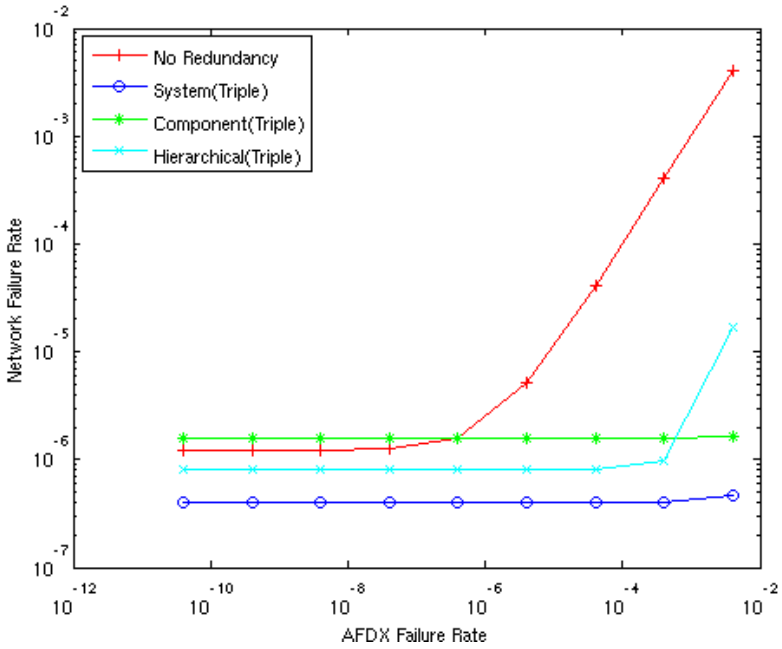


Figure 4-11: Architecture‟s unreliability as a function of the AFDX‟s unreliability for data correction

Figure 4-10 and Figure 4-11 confirm that the reliability of the various architectures is almost equivalent when considering the error detection and data correction redundancy management scheme. The very small differences between the 2 redundancy schemes can here again be explained by the reliability of the coupling elements. The reliability of the coupling elements then becomes fundamental to ensure the effectiveness of the reliability increase based on modular redundancy. The level of redundancy can be used to change the scope of the error management approach, but is not efficient in lowering the overall network reliability. For low reliability of the coupling elements, higher levels of redundancy might then be useless for a significant increase in complexity.

Figure 4-12 and Figure 4-13, explore the relationship between the reliability of the studied architectures and the coupling elements. Several reference points in the form of vertical dotted lines have been included in the figures to allow an easier analysis of the architectures behavior. The DMR stands for a duplication of the voting system to reduce the coupling‟s effect on the system, while TMR implies triplication of the voting system. Although the coupling element unreliability is not limited to these lines, these reference points represent values of unreliability that can be achieved through modular redundancy based on our specified unreliability value for

the coupling elements (when using the failure rate value of $4*10^{-9}$). Although several works have already presented how to improve the reliability of the voting system [62, 80, 81], our analysis focus on the effects and needs for better voting mechanisms. This can be contrasted to others means of obtaining more reliable systems such as selecting more reliable components.

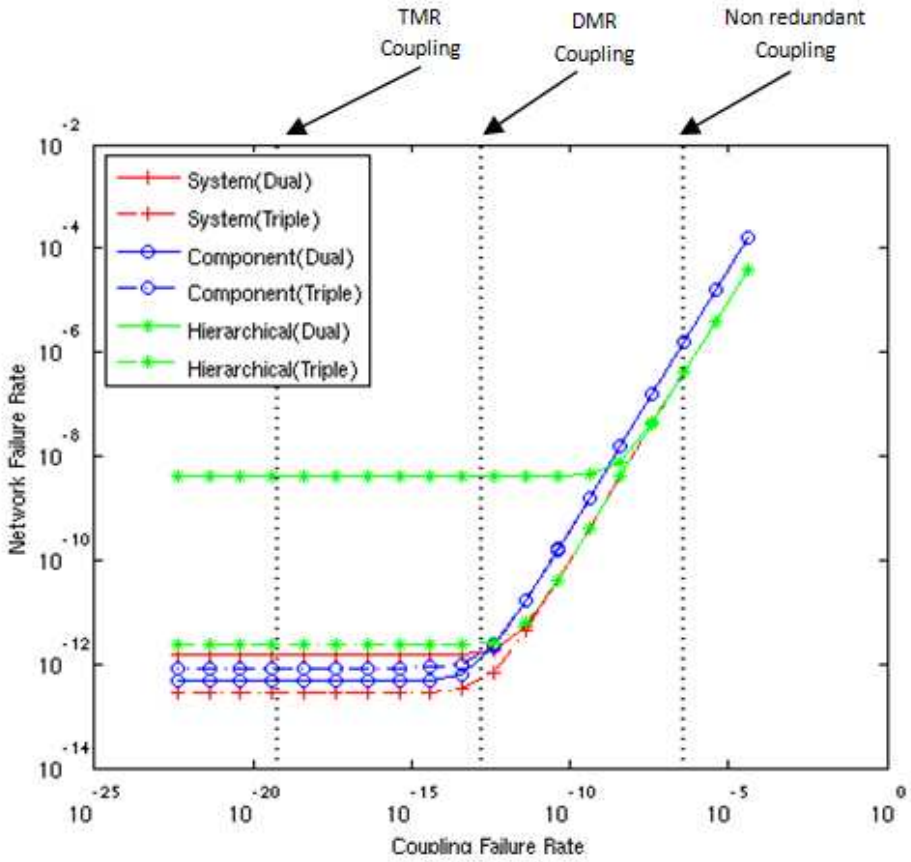


Figure 4-12: Architecture‟s unreliability as a function the coupling‟s unreliability for error detection

We can see from both figures that an improvement of the reliability of the coupling elements greatly benefit all the studied architectures up to a certain point, where the system‟s reliability becomes independent of further improvements in the coupling‟s reliability. The reliabilities of the presented architectures become dominated by the reliability of the non-redundant layers as shown by the existence of plateaus in both figures. For the dual redundancy implementation of the hierarchical architecture, the reliability plateau of 10^{-7} corresponds to the reliability of the non-duplicated AFDX layer. For the other architectures, each plateau corresponds to the reliability of the limiting layer thus granting a network reliability of the order of 10^{-12}, because of the dual

nature of the redundancy. The limiting layer can be defined as the layer with the lowest reliability.

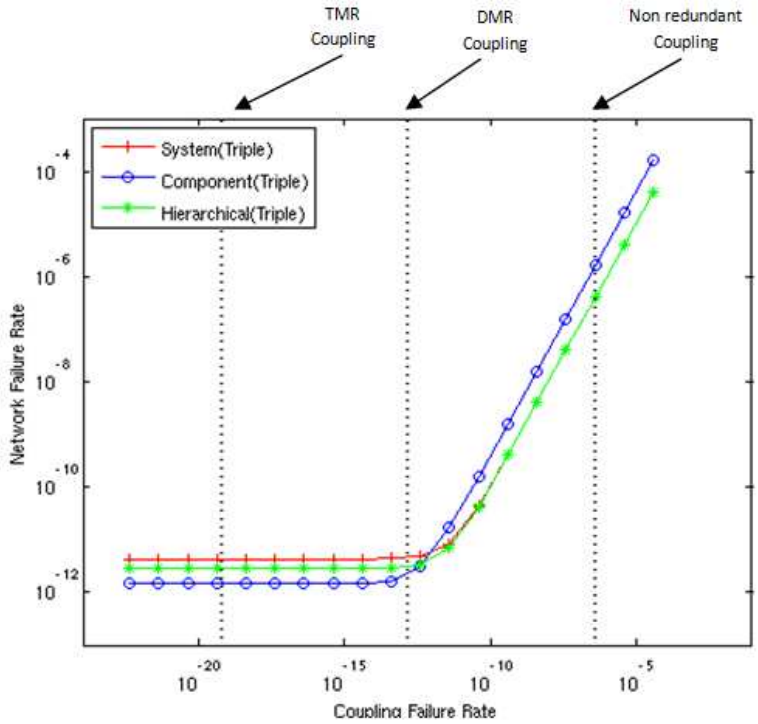


Figure 4-13: Architecture‟s unreliability against the coupling‟s unreliability for data correction

In order to benefit from the hierarchical architecture, the reliability of the coupling factor must then be lesser than or equal to that of the limiting layer, the AFDX layer in our case study. For example, we see in Figure 4-12 and Figure 4-13 that the hierarchical architecture offers the best reliability when considering our typical non-redundant voting system, but reaches a limit, denoted by the plateaus, for more reliable versions of the coupling elements. Unlike the other architectures, a reliability improvement of the coupling over the reliability of the limiting layer does not reflect into a reliability improvement of the whole network. In this last case, improving the reliability of the voting layer would be a waste of time and resources, since it would not have any significant effect on the system‟s reliability. In Figure 4-13, triplication of the voting system would result in the same waste as it does not improve the system‟s reliability. With the help of these figures, it is possible to evaluate the effectiveness of the selected architecture and redundancy management scheme in regards to the reliability of its basic components and of the coupling elements. Depending on the application, the generic nature of the hierarchical

architecture allows to easily match another set of requirements without having to do a complete system redesign. Although the redundancy levels of our test case architecture are very similar, due to the reliability of our implemented components, our architecture is well suited for the avionic domain. Our hierarchical architecture would then easily be applicable to a system similar to Figure 4-2, in which the number of redundant ARINC 429 busses required to connect all the duplicated components could be reduced. The system and component architectures could also be used to re-implement the same system using the IMA concept, but that would be less efficient than with the hierarchical architecture. A system architecture in which the same level of redundancy is applied at each layer would not be as flexible as the hierarchical architecture. As for the component architecture, the number of required voters inducing single points of failure would be much higher than with the proposed hierarchical architecture resulting in lower system reliability. The proposed hierarchical architecture takes advantage from the best features of the component and system architectures. It allows reducing the number of voters and provides the flexibility to implement each layer with the best level of redundancy.

4.6 Conclusion

In this paper, we presented an approach to integrate redundancy into the design of an avionics network. The approach is based on the utilization of a hierarchical architecture allowing the appropriate inclusion of redundancy at any specified layer. Benefits and limitations of this architecture have been presented in comparison to traditional redundancy management schemes. Unlike previous findings, our analysis led us to conclude that the component architecture, which is supposed to provide the optimal reliability in most applications, can be ill-suited for avionics systems, because of the presence of coupling dependencies creating single points of failures. The presented hierarchical architecture provides reliability similar to the system redundancy approach, while consuming less resources. For our case study of a sensor network implemented on a hardware platform, we were able to reach the reliability targets while reducing the number of individual local components and a complete AFDX network, which can represent a considerable gain in resource and design effort. Analysis of the sensitivity of the architecture shows that the coupling elements, such as the required voting system, become the main limit to improving reliability through redundancy. Through our analysis, the required reliability of the voting system can be determined, thus eliminating unnecessary design efforts and resources. In conclusion, we

demonstrated in this paper the benefits and limitations of a hierarchical approach to redundancy management.

CHAPITRE 5 DISCUSSION GÉNÉRALE

5.1 Architecture générique de réseau avionique

5.1.1 Sommaire

Avec l'avènement anticipé de systèmes plus performants dans le domaine avionique, une quantité grandissante de capteurs et d'actuateurs sera nécessaire pour répondre aux fonctionnalités requises dans les systèmes de prochaine génération. Afin de faciliter l'intégration de tous ces capteurs et actuateurs aux systèmes avioniques, nous avons proposé une architecture hiérarchique de réseaux de capteurs s'appuyant sur le modèle de base de la norme IEEE 1451. On retrouve en effet dans notre architecture les modules de base de la norme IEEE 1451 (TIM et NCAP) malgré l'ajout de mécanismes de gestion de la redondance afin de pouvoir améliorer la fiabilité du réseau. N'étant aucunement spécifié dans la norme, nous avons introduit notre propre gestion de la redondance permettant entre autres de modifier le niveau de fiabilité selon les exigences. L'architecture proposée peut donc être maintenant utilisée dans le cadre d'applications critiques en plus de conférer un caractère réutilisable et une meilleure consommation de ressources par rapport aux architectures avioniques traditionnelles. De manière à respecter les différentes contraintes liées au développement de systèmes avioniques, l'architecture proposée permet la personnalisation de la bande passante offerte, du niveau de fiabilité recherché, de la quantité de ressources consommées et du type de capteurs ou actuateurs utilisés. De manière à quantifier les gains dus à l'utilisation de notre architecture, deux architectures traditionnelles sont d'abord présentées, soit l'architecture dite de base, se rapprochant de l'architecture fédérée, et une architecture distribuée, utilisée dans de nombreux domaines moins critiques que l'avionique. La modélisation de plusieurs configurations est présentée dans l'article du chapitre 3 afin de pouvoir comparer les différentes architectures et d'en faire ressortir les avantages et inconvénients. Le processus de validation des configurations retenues ainsi que les analyses pertinentes reposent sur l'implémentation d'un réseau de capteurs développé en suivant les conventions avioniques. Le prototype développé utilise 2 cartes FPGA LX45T de Xilinx pour une première connexion de capteurs à un bus secondaire respectant la norme ARINC 825. Les données transmises sur ce bus secondaire sont par la suite regroupées et envoyées vers les ordinateurs de contrôle de vols à travers un réseau primaire de type AFDX. Au niveau de la fiabilité, nos analyses montrent la

polyvalence de notre architecture en permettant la modulation de la fiabilité totale du réseau grâce à la variation intégrée du nombre de composants ou de bus, contrairement aux approches traditionnelles qui ne possèdent aucun mécanisme intrinsèque pour augmenter la fiabilité du réseau, ce qui rend leur application inefficace dans un contexte d'applications critiques. Au niveau de la consommation ressources, notre architecture permet des gains raisonnables lors d'une application directe sans contrainte explicite de fiabilité. Bien que des gains soient observés par rapport aux deux architectures de comparaison, les gains face à l'architecture basique sont substantiellement plus importants que ceux réalisés par rapport à l'architecture distribuée. Dans un cadre d'applications critiques, les écarts de consommation de ressources observés démontrent encore une fois la supériorité de notre architecture, directement liée à notre gestion de la redondance face à une simple application de la redondance. La section suivante présente une discussion sur l'article du chapitre 3 contenant le détail des travaux et analyses énoncés ci-dessus.

5.1.2 Discussion

La présente section revient plus précisément sur les contributions mentionnées dans l'article du chapitre 2 en venant les situer et en apportant quelques explications complémentaires aux propos déjà publiés. Pour commencer, nous sommes les premiers à proposer une architecture permettant l'intégration de capteurs intelligents dans une application avionique critique. Tel que montré dans notre revue de littérature, plusieurs chercheurs [12] [14] [17] [88] [89] [16] [90] ont utilisé des capteurs intelligents ou une intégration respectant la norme IEEE 1451. L'emploi d„IEEE 1451 dans leurs travaux se résumait par contre au niveau des tests ou de la programmation au sol. Nous n'avons trouvé aucune référence à une application réelle de la norme IEEE 1451 dans l'implémentation d'un système avionique bien que son utilisation dans le domaine avionique peut entraîner de nombreux avantages selon les experts de la NASA [15].

La présence de différents niveaux hiérarchiques permet une optimisation en fonction des ressources consommées ou du niveau de fiabilité. Les différentes contraintes viennent dicter les compromis architecturaux, ce qui s'avère particulièrement utile dans un contexte de reconfiguration dynamique. À l'aide de la plateforme matérielle présentée dans le chapitre suivant et modélisant un système avionique fixe, plusieurs configurations peuvent être implémentées selon les contraintes exigées. Un système non critique pourrait éventuellement demander un niveau de fiabilité plus élevé durant certaines phases de vol ou en raison d'une mise

à jour complète du système. Une réduction du nombre de ressources consommées pourrait même entraîner une baisse de la consommation de puissance. L'accommodation dynamique des systèmes avioniques est un bon exemple de nouvelles capacités rendues possibles par l'architecture IMA mais n'étant pas adaptée aux techniques de design usuelles. Afin d'être applicable dans le développement de systèmes IMA, une plateforme pouvant valider les différentes configurations devient donc nécessaire et motive en partie le développement de la plateforme de design et validation présentée dans le chapitre suivant.

En plus de pouvoir accommoder la connexion de capteurs intelligents à travers une architecture hiérarchique, le niveau de ressources entraîne une consommation réduite de ressources matérielles. À travers notre implémentation, nous avons déterminé que notre architecture consomme jusqu'à 87% moins de ressources qu'une architecture traditionnelle pour un réseau de capteurs intelligents de fiabilité similaire. Cette architecture est fidèle au concept antérieur à l'IMA, soit l'approche fédérée, démontrant à nouveau les avantages de la dernière évolution. Le gain est plus mitigé en comparaison à une configuration distribuée conforme à la norme IEEE 1451 dans des applications non critiques. Dans certains cas de figure d'applications critiques, nous avons par contre démontré que notre architecture pouvait entraîner une hausse de fiabilité importante par rapport à l'architecture distribuée de la norme IEEE 1451. En raison de ces premières analyses, il nous est apparu important d'également inclure les points communs de défaillances motivant, entre autres, les travaux présentés dans l'article du chapitre 4. La prochaine section poursuit avec la plateforme matérielle de design et validation utilisée dans le développement de l'architecture hiérarchique présenté dans l'article du chapitre 3.

5.2 Validation de l'approche mixte pour l'intégration de réseaux avioniques

5.2.1 Sommaire

La constante évolution des systèmes avioniques requiert toujours l'adoption de nouvelles technologies de communication de données, comme ce fut le cas pour les protocoles ARINC 825 et ARINC 664 introduits il y a déjà quelques années. Les caractéristiques et performances peuvent varier énormément d'une norme à l'autre. L'intégration de ces protocoles peut donc demander un effort considérable afin de respecter les différentes contraintes du domaine

avionique. Bien que des composants commerciaux soient disponibles, l'utilisation d'une implémentation matérielle personnalisée et générique peut s'avérer extrêmement utile dans la validation d'architectures ou d'analyses théoriques et l'identification des cas problématiques. À travers une plateforme de prototypage de systèmes avioniques et une architecture générique, nous proposons dans la prochaine section qui suit une approche de design et validation de tels systèmes. La plateforme matérielle permet, entre autres, d'injecter des pannes et de mesurer la latence des trames à travers le réseau. Les différentes contraintes de performance et de fiabilité peuvent être mesurées pour satisfaire les nombreuses normes de certification et de design. Notre prototype se base sur deux FPGA de type Spartan-6 développé par Xilinx. Pour le cas étudié dans les sections suivantes, la plateforme est configurée pour représenter un réseau de transducteurs pouvant garantir une bande passante de 1 Mbit/s à travers un bus redondant de type ARINC 825. Les transducteurs peuvent, par la suite, être connectés à un ordinateur à travers une connexion AFDX pour traitement des données et validation. À l'aide d'une meilleure connaissance et une approche structurée telle celle présentée dans la prochaine section, les processus de design, validation, intégration, certification et d'entretien de systèmes avioniques complexes et critiques peuvent grandement en bénéficier. Les détails de l'approche proposée, la plateforme matérielle ainsi que leurs avantages et limites sont détaillés dans la dernière partie de cette section.

5.2.2 Méthodologie

En se basant sur la littérature présentée à la section 2, nous avons identifié le besoin de pouvoir valider le design de systèmes avioniques complexes et l'intégration de nouvelles technologies de manière systématique et plus efficace. Afin de répondre à ce besoin, nous présentons donc notre approche développée dans le but de valider une nouvelle architecture de réseau de capteurs à bord d'un avion. La prochaine section vient brièvement présenter les étapes que nous avons suivies. La Figure 5-1 présente la méthodologie globale de notre approche.

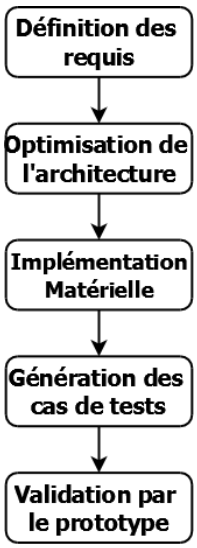


Figure 5-1: Méthodologie proposée

Lors du développement de notre approche visant les applications critiques en avioniques, nous avons tenu à prendre en considération le processus de certification requis lors de la conception de composants avioniques. Bien que notre approche matérielle ne requière l'application que du processus de certification DO-254, les mêmes principes sont applicables à une implémentation logicielle reposant sur d'autres normes de certification. En nous basant sur les conclusions d'un groupe d'ingénieurs de la société Mentor Graphics portants sur l'application de la norme DO-254 [91], plusieurs caractéristiques ont été identifiées afin de réduire les défis de vérification qu'elle impose. Pour une approche de design spécifique, les 5 caractéristiques suivantes ont été identifiés comme étant les plus importantes :

- Conformité par rapport aux examens particuliers du DO-254
- Utilisation des normes de l'industrie au niveau des outils et des langages de programmations sélectionnés
- Compatibilité avec le design actuel ainsi que toute version subséquente
- Support des nouvelles structures de vérification matérielle
- Utilisation de cas de test facile à créer, maintenir et modifier

L'architecture générique combinée avec la plateforme matérielle réutilisable permet de satisfaire à elles-seules la plupart des caractéristiques énoncées ci-haut. En se basant sur ces dernières caractéristiques, notre approche permet donc de réduire la complexité du processus de

certification tout en améliorant l'efficacité globale du processus de design de composants avioniques. Chacune des étapes de notre approche sera maintenant brièvement présentée en explicitant au passage les liens avec les caractéristiques précédentes.

La première étape de notre approche proposée pour le design de réseaux avioniques ou pour l'intégration de nouveaux protocoles de communication repose sur la définition des requis. Cette étape est particulièrement cruciale étant donnée la quantité importante et le caractère contraignant des règlements du domaine en particulier dans l'atteinte des cibles de fiabilité. Dans le contexte de nos travaux, nous nous sommes fixé des requis au niveau de la fiabilité du réseau, du déterminisme, de la latence maximale des trames, de la charge sur le réseau ainsi que la bande passante atteignable de manière à respecter des contraintes réalistes basées sur les attentes de nos partenaires industriels du projet AVIO 402. Notre approche tente de satisfaire de manière simultanée toutes les contraintes bien que des compromis peuvent manuellement être sélectionnées lorsqu'une architecture ne peut répondre à toutes les contraintes en même temps. Bien que nous nous soyons limité aux requis énoncés dans cette section, plusieurs requis supplémentaires, tels la consommation de puissance, le poids du système, le nombre de nœud du réseau ou la longueur des fils, pourraient facilement venir intégrer notre approche.

La seconde étape repose sur la description de l'architecture par des modèles MATLAB. Pour un choix donné au niveau de la configuration du réseau en définissant le nombre de bus, de TIMs et de NCAPs, les performances du réseau sont obtenues analytiquement et comparés par la suite avec les contraintes sélectionnées. À travers nos travaux, nous avons toujours été en mesure de générer une configuration respectant les requis imposés bien que la quantité de ressources nécessaires puissent varier d'un cas de figure à l'autre. Plus les contraintes sont sévères, plus la consommation de ressource tends habituellement à s'élever également. Ce processus est maintenant effectué de manière manuelle mais son automatisation permettrait d'obtenir un choix de configurations plus optimales dans un laps de temps plus court. Cette automatisation pourrait entre autres faire l'objet d'une prochaine étape dans la continuation de ce projet. Le prochain tableau vient présenter les requis sélectionnées pour l'atteinte des objectifs de notre projet.

Tableau 5.1: Requis

Requis	Contrainte
Taux d'échec	$< 10e^{-6}$
Charge	< 50%
Déterminisme	Complet
Latence Maximale	< 2ms
Bande Passante	1 Mbit/s

En terme de fiabilité, nos analyses reposent sur le MTBF des différents composants de base tel que propose dans le manuel militaire MIL-HDBK-217 [75] couramment utilisé dans le domaine avionique. Les contraintes de charge maximale et de déterminisme sont imposées par la norme ARINC 825 qui limite à 50% la charge de chaque bus en opération normale. La charge sur un bus ARINC 825 est obtenue par l'équation suivante :

$$Charge = \frac{\sum(Taille\ de\ trame * Nb\ de\ trame)}{Intervalle\ de\ transmission * Bande\ Passante} \quad (5.1)$$

L'étape suivante consiste à implémenter les modules de base servant à la réalisation de la configuration déterminée à l'étape précédente. L'utilisation d'une architecture générique permet entre autres l'implémentation des modules selon un choix variée de méthodes, tel une approche matérielle, logicielle ou basée sur des composants propriétaires déjà implémentés. Notre implémentation repose la description matérielle de tous les composants à l'aide du langage VHDL. Notre implémentation permit entre autres de tester et valider un mécanisme de gestion des erreurs proposé par [92] à l'aide de la reprogrammation des FPGAs. Le développement complet des modules nous offre en effet le plein contrôle sur la sélection et la configuration de ces mécanismes de gestions de pannes. À ce point, nous avons également implémenté les interfaces réelles de capteurs typiques ainsi que des modules d'émulation de trafic à haut niveau pour améliorer la phase de test et de validation effectué à l'aide du prototype. Plus de détails sur l'implémentation résultante de notre projet seront exposés dans la section suivante.

La quatrième étape sert à définir et implémenter les cas de test utilisés afin de valider l'architecture sélectionnée ainsi que l'implémentation et la connexion des différents composants.

La reconfiguration du prototype et le recours à une architecture générique permet entre autres de pouvoir valider simplement plus d'une architecture à l'aide de la même plateforme. Plusieurs mécanismes d'injection de pannes ont été ajoutés dans les modèles utilisés pour l'optimisation de l'architecture. Ces modèles servent par la suite à définir les scénarios des pires cas pour la génération de cas de tests. Ce dernier aspect est particulièrement utile dans le support de processus de certification. Finalement, la planification pour l'inclusion de mécanismes de mesures des différentes contraintes doit être effectué afin d'extraire les performances du prototype. Ce dernier point est la motivation derrière le développement de notre propre système de monitorage de la latence des trames sur notre réseau. Bien que notre réseau repose en partie sur le protocole ARINC 825 au niveau du bus secondaire, notre approche diffère des travaux présentés au chapitre 2 en mettant le focus sur la mesure globale de la latence sur l'ensemble du réseau plutôt que sur le bus lui-même. Ces approches de monitorages traditionnels à plus bas niveau peuvent mener à de mauvaises interprétations de l'impact de certains comportements au niveau du système, tel la gestion incorrecte du système de gestion des erreurs dans notre implémentation.

La dernière étape de notre approche consiste à l'assemblage et la programmation de la plateforme matérielle de prototypage et à l'application des cas de tests définis précédemment. Même à travers un prototype réduit comme le nôtre, nous sommes quand même capables d'appliquer les cas de test désirés en ayant recours entre autres à des modules d'émulations à plus haut niveau. Ce dernier aspect est particulièrement utile lors de la certification puisque l'exploration de tous les cas possibles est habituellement requise dans le cas de système complexe. En combinant l'implémentation initiale avec le processus de certification, l'effort total nécessaire pour le développement de composants avionique s'en trouve réduit. Le processus de reconfiguration et de programmation des FPGAs s'effectue également à ce niveau de manière manuelle mais pourrait grandement bénéficier d'une approche automatisé faisant suite à l'étape d'optimisation de l'architecture.

5.2.3 Implémentation

La présente section servira à présenter les détails ayant mené à l'implémentation réussie avec succès du prototype développer dans le cadre du projet AVIO 402. Au niveau des requis, nous concentrer sur les requis énoncés à la section précédente. Au niveau de l'optimisation de

l'architecture, nous commençons par présenter les analyses de fiabilité qui ont servi à s'assurer du respect des contraintes de notre prototype. L'évaluation de la fiabilité de l'architecture devient essentielle dans l'achèvement du processus final de validation des réseaux avioniques. Pour cette étape d'optimisation d'architecture, nous avons eu recours à certaines des techniques énoncées à la section 2.2.1, soit la modélisation mathématique des pannes et les arbres de pannes (FTA).

En nous basant d'abord sur la modélisation de fiabilité traditionnelle et le recours aux arbres de pannes, nous avons d'abord conçu des diagrammes hiérarchiques représentant chacune des parties du réseau. Ces diagrammes ont ensuite été convertis en équations permettant l'évaluation rapide et en phase préliminaire de multiples configurations regroupant des éléments de base. Dans les équations suivantes (5.2 à 5.10), R représente la probabilité de succès ou la fiabilité de l'item associé, tandis que Q correspond à la probabilité d'échec du même composant, suivant la relation suivante :

$$R = 1 - Q \qquad (5.2)$$

Chaque diagramme d'arbre de pannes est, par la suite, accompagné des équations correspondantes. À l'aide de ces diagrammes et équations, la probabilité de corruption causant une lecture incorrecte est évaluée. La Figure 5-2 présente le diagramme FTA du plus haut niveau de l'architecture pour le type de réseau sélectionné. Les indices dans les équations font référence aux composants de figures 5-2 à 5-4 qui sont résumés dans le Tableau 5.2. Les équations 5.3 à 5.9 qui suivent les figures 5-1 à 5-3 ont été obtenues grâce à la modélisation des arbres de pannes associés. Dans cet exemple de modélisation, il est important de noter que le « gateway » AFDX ou toute partie du réseau AFDX ne sera pas inclus dans cette discussion, puisque l'influence du réseau AFDX dans la fiabilité du système global est présenté dans l'article du chapitre 4 [9]. L'implémentation logicielle de la couche AFDX entraîne également une divergence au niveau de l'estimation de la fiabilité face au reste de l'implémentation matérielle du réseau.

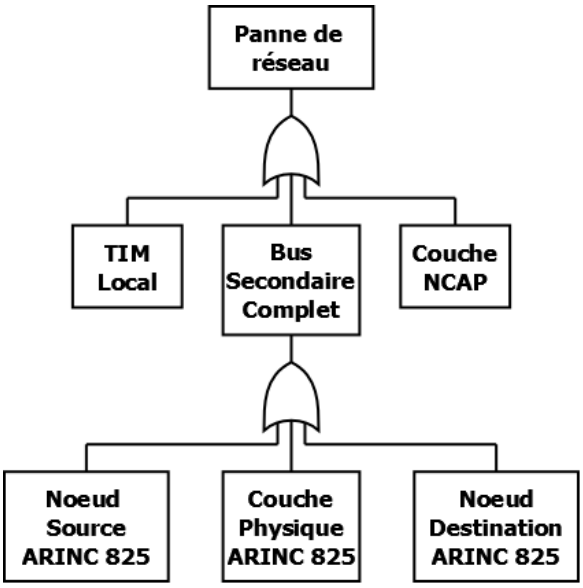


Figure 5-2: Arbre de panne du réseau complet

$$R_{Réseau} = R_{TIM}\ R_{RS}\ R_{NCAP_L} \qquad (5.3)$$

$$R_{FB} = R_{825_I}\, R_{825_CP}\, R_{825_I} \qquad (5.4)$$

Tableau 5.2: Notation adoptée pour décrire les composants

Composant	Étiquette
Réseau Complet	Réseau
Perte d'alimentation	PA
Réseau secondaire complet	RS
Couche physique ARINC 825	825_CP
Interconnexion ARINC 825	825_I
Couche NCAP	NCAP_L
NCAP	NCAP
Cœur NCAP	NCAP_C
TIM	TIM
TMI	TMI
Cœur TMI	TMI_C
Interconnexion directe de transducteur	TMI_I
Couche de service	SERV
Couche de communication	COM
Commutateur (« Crossbar »)	C

La Figure 5-3 et la Figure 5-4 présentent respectivement les arbres de panne pour un TIM et un NCAP, suivis par leurs équations de fiabilité associées. Il est important de noter que les TIM et

NCAP font partie des composants de base du diagramme de niveau supérieur, étant donné la nature hiérarchique de l‟architecture.

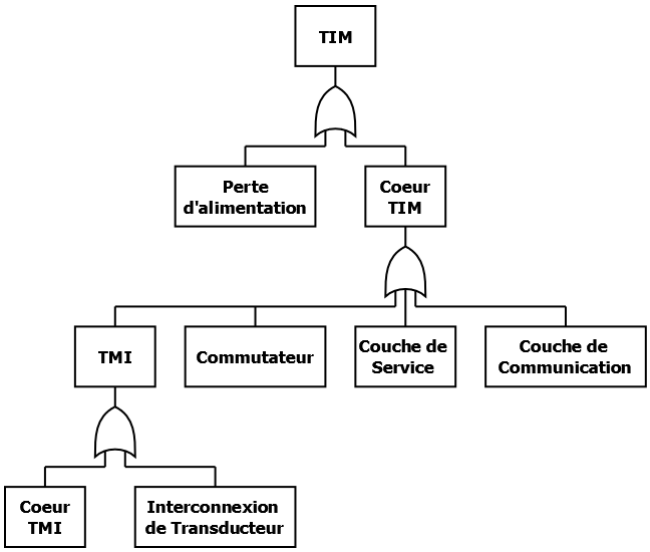


Figure 5-3: Arbre de panne d‟un TIM

$$R_{TIM} = R_{TIM_C}\ R_{PA} \qquad (5.5)$$

$$R_{TIM_C} = R_{C1}\ R_{SERV}\ R_{C2}\ R_{COM}\ R_{TMI} \qquad (5.6)$$

$$R_{TMI} = R_{TMI_I}\ R_{TMI_C} \qquad (5.7)$$

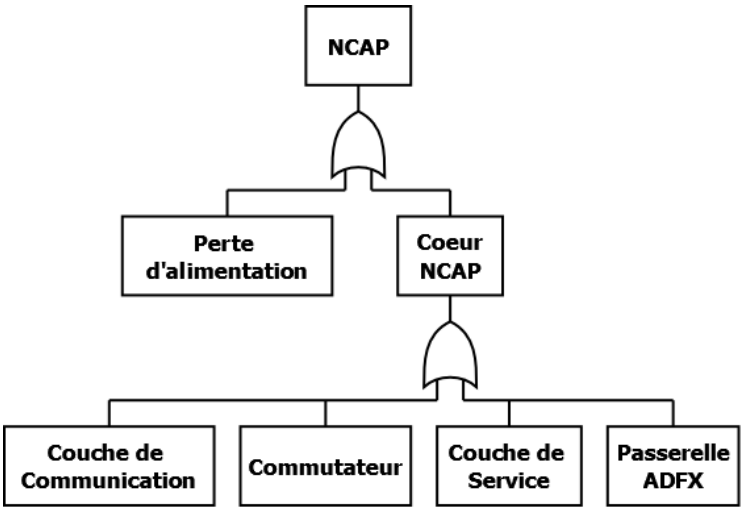


Figure 5-4: Arbre de panne d‟un NCAP

$$R_{NCAP_C} = R_{COM}\ R_{C}\ R_{SERV} \qquad (5.8)$$

$$R_{NCAP} = R_{NCAP_C}\, R_{PA} \qquad (5.9)$$

Plusieurs des composants de l'architecture sélectionnée sont organisés dans une configuration parallèle où un nombre minimal de composants doit toujours être fonctionnel afin d'assurer la fonctionnalité globale du système. Ces structures sont identifiées dans les diagrammes ci-dessus avec le terme « layer » signifiant couche. À la Figure 5-4, l'évaluation de la fiabilité de la couche NCAP est basée sur le nombre minimal de NCAP requis par rapport au nombre total de NCAP disponible. L'équation 5.10 tirée de [87] exprime la fiabilité de ce type de structure, où K représente le nombre de composants fonctionnels tandis que N représente le nombre total de composants. Dans le cas présent, le taux d'échec de chaque composant appartenant à une même couche est considéré comme identique. Cette structure et l'équation associée ont également été utilisées pour l'estimation de rendement lors de la production de circuits intégrés.

$$R(M) = \sum_{x=K}^{M} \frac{N!}{x!\,(N-x)!} R^x Q^{N-x} \qquad (5.10)$$

Pour en arriver à une analyse complète, les effets des interconnexions et des pannes d'alimentation du système ont également été inclus. Chaque trame traverse donc le réseau d'un nœud source vers un nœud de destination tel que présenté à la Figure 5-2. Pour l'analyse du réseau secondaire ARINC 825, les estimations de fiabilité prennent en considération tous les types possibles de défaillances. Pour la modélisation des pannes d'alimentation, chaque TIM et NCAP nécessiterait sa propre alimentation selon les systèmes avioniques traditionnels. La présence de composants de type alimentation est incluse pour indiquer la perte totale d'un TIM ou NCAP suite à une panne de la distribution de l'alimentation. Bien que les diagrammes et équations présentés dans cette section soient basés sur notre implémentation d'un réseau avionique de capteurs, ce type d'analyse sert également pour tout type de système dans plusieurs domaines. Suite à l'évaluation de la fiabilité et l'obtention d'une configuration répondant aux contraintes, nous pouvons donc passer à l'étape d'implémentation.

Afin d'implémenter l'architecture sélectionnée, nous avons déterminé qu'au moins 2 cartes FPGA étaient nécessaires à l'application des cas de tests sélectionnés. Notre prototype est donc basé sur 2 cartes d'évaluation SP605 produit par Xilinx [93]. Ces cartes ont été choisies entre autres pour leurs nombreuses interfaces de connexion facilitant la portabilité et toute extension éventuelle. La complexité au niveau ressources des différents modules composant notre prototype

finale est exposée dans le Tableau 5.3. Les résultats de ce tableau monte en effet la quantité requise de composants à implémenté sur chacune des cartes peut facilement y être contenue. Comme notre implémentation finale de notre TIM laisse de nombreuses ressources toujours disponibles, une reconfiguration locale pourrait donc permettre de desservir aisément de nouveaux capteurs. Les résultats de consommation totale de ressources repose sur l'utilisation de 4 FPGA qui seraient minimalement requis pour l'implémentation réelle de l'architecture finale.

Tableau 5.3: Complexité des composants

Module	LUTs		Registers	
Contrôleur ARINC 825 simple	856	2%	1101	4%
Contrôleur ARINC 825 double	1831	3%	2126	8%
Module de service du TIM's	5975	11%	11198	41%
Module de service du NCAP's	1112	2%	850	3%
TIM (4 capteurs)	7806	14%	13324	49%
NCAP	2943	5%	2976	11%
Réseaux entiers	21498	10%	24522	11%

Au niveau de l'implémentation de la couche physique du réseau secondaire ARINC 825, notre prototype a recours à quatre connecteurs CAN traditionnels se trouvant sur les deux cartes de réseautage ISM [94]. Le prototype dans une configuration à un TIM et un NCAP est présenté à la Figure 5-5. Dans cette image, plusieurs capteurs de température COTS sous un même assemblage sont connectés à un même TIM. Pour la phase de validation, nous avons eu recours à 4 de ces capteurs à travers une interface suivant le protocole I2C. Les capteurs additionnels peuvent être connectés à d'autres TIM pour des cas de tests en nécessitant plusieurs. À l'aide de la reconfiguration et la connectivité offertes par la plateforme FPGA, de nombreux types de transducteurs, modèles et émulateur peuvent se connecter dès la phase initiale de validation. Dans le cadre de notre projet, nous servons également à valider la conception de nouveau type prometteur de capteurs et le nouveau design de module analogique d'acquisition des données.

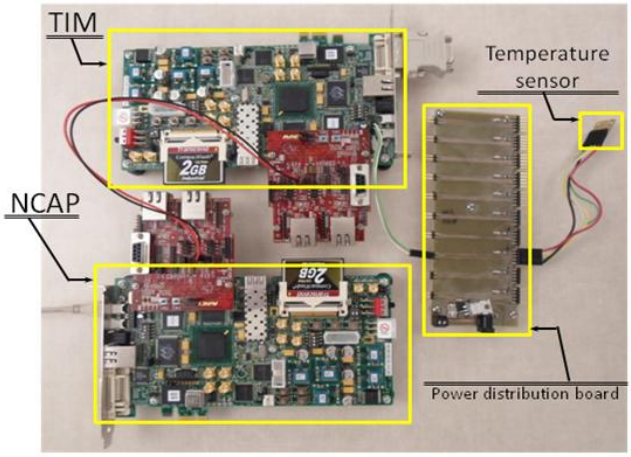


Figure 5-5: Prototype matériel

Dans le contexte de notre projet, notre implémentation de la norme IEEE 1451 s'est limitée aux fonctions minimales pour assurer la communication et le traitement simple des données. Cette approche d'implémentation réduite lors d'une première implémentation est même recommandée par un comité d'experts de la NASA [95]. De leur point de vue, une implémentation de base peut toujours être augmentée de nouvelles fonctionnalités tous en demeurant compatible avec tout type de capteurs. En combinaison avec l'utilisation de notre approche, ce dernier aspect ne peut qu'accentuer la réduction de l'effort lors des phases initiales de design et validation. Des mécanismes d'injection de pannes sont également inclus dans notre prototype afin de valider la gestion des pannes de notre système, telle la gestion de la redondance du contrôleur ARINC 825 double.

De manière à garantir la nature générique de l'architecture, notre implémentation comprends l'identification de contrôleur défectueux et rediriger le trafic à travers les contrôleurs additionnels encore en opération. Au niveau du contrôleur ARINC 825 double, le mécanisme de gestion de la redondance permet la transmission de toutes les trames sous la contrainte de latence en contrôlant les queues d'entrées de chaque contrôleur individuel. À l'observation d'une dégradation sur un bus, le trafic est redirigé vers les bus fonctionnels. Notre mécanisme de gestion se base sur le statut interne de chaque contrôleur qui entraîne même la fermeture complète d'un bus après des perturbations consistantes. Le cout additionnel de ressource pour mécanisme de gestion est très limité par rapport à un seul contrôleur, sans même considérer l'apport additionnel dû à la redondance.

Bien que notre implémentation se situe à faible niveau d'abstraction, nos requis demandaient la couverture du système dans son ensemble. En se basant sur ce principe, nous avons dû implémenter notre propre mécanisme de mesure de la latence puisqu'aucun outil ne permettait la mesure du système dans son ensemble. Ce moniteur nous permet donc de mesurer aisément en temps réel l'impact de changement de protocole, de stratégie de gestion de pannes ou toute autre modification. En gardant la même structure de test, les mêmes requis peuvent être validés pour de nombreuses architectures différents avec un effort minimale s'assurant même des contraintes de certification. Le mécanisme de mesure repose sur l'inclusion dans chaque trame de deux champs additionnels contenant les valeurs de latence de la trame associée. Ces deux champs sont inclus directement dans la trame et correspondent au temps en transit dans le TIM et NCAP respectivement. Comme le TIM et le NCAP ne sont pas implémentés sur la même carte physique, la synchronisation des deux cartes est requise. Notre moniteur se sert donc du mécanisme assurant le synchronisme entre les nœuds ARINC 825. Dans notre système, cette synchronisation est également utilisée par nos modules de mesure. Cette technique induit malheureusement une augmentation de la consommation de bande passante. De manière à compenser cet inconvénient, une correction mathématique doit être appliquée pour obtenir la latence exacte. En se référant à la littérature du chapitre 2 [40], cette méthode de mesure permettant la mesure sur plusieurs cartes matérielles à travers différents domaines de temps diffère des approches traditionnelles à ce niveau.

La dernière étape de l'approche proposée consiste en la validation du respect des contraintes de deux architectures distinctes. Les résultats présentés viennent présenter la validation de deux architectures par rapport aux mêmes contraintes énoncées plus dans cette section. La bande passante des différents réseaux, fixée à 1 Mbit/s pour le réseau secondaire ARINC 825, est confirmée à l'aide de l'outil ADS2 permettant la simulation de protocoles avioniques. Les mesures de trames moyennes présentées ci-contre ont été obtenues sur in intervalle de transmission de 1 sec correspondant à 500 cycles de captures de données. Le reste de la méthodologie de test est la même que dans la méthodologie présenté précédemment.

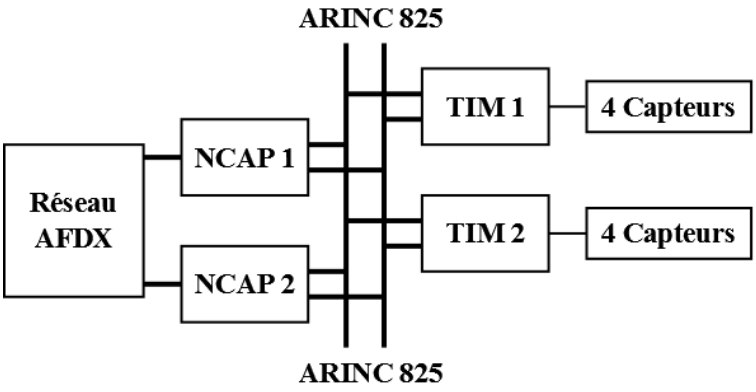


Figure 5-6: Exemple de réseau

Les mesures de latence du Tableau 5.4 sont présentées pour deux configurations du même réseau de manière à faire ressortir les compromis entre fiabilité et performance. La première configuration (A), présentée à la Figure 5-6, a été sélectionnée de manière à respecter toutes les contraintes imposées, tandis que la seconde configuration (B), reposant plutôt sur un seul bus secondaire, est simplement utilisée comme comparaison.

Tableau 5.4: Latence moyenne de trames

Capteur	Latence (us)	
	Configuration A	Configuration B
1A	337	337
2A	417	425
3A	552	505
4A	636	617
1B	337	752
2B	449	857
3B	554	940
4B	638	1024

On peut observer à partir du Tableau 5.4 qu‟une latence minimale inférieure à 2 ms est aisément atteignable avec la configuration A pour le nombre de capteurs considéré. La variation de latence observée entre les différentes trames redondantes est relativement faible et causée par le remplissage de bit du protocole ARINC 825. Puisque les trames respectent toujours le même ordre de transmission et de réception avec une légère variation de latence, on peut conclure au déterminisme du réseau.

Tableau 5.5: Validation de réseau

Requis	Configuration	
	A	B
Taux d'échec	6.738E-07	2.1957E-06
Charge	33%	53%
Déterminisme	Full	Full
Latence Max (ms)	0.65	1.05
Bande Passante	2 Mbit/s	2 Mbit/s
Complexité (LUT)	17898	12231

Le Tableau 5.5 résume les caractéristiques plus précises des deux mêmes configurations. Les données du Tableau 5.5 ont été obtenues en fixant la valeur échantillonnée par le capteur de manière à générer le plus grand nombre de bits additionnels. On obtient donc ainsi le cas du pire scénario possible, tout en faisant disparaître toute variabilité des temps de transmission. On peut voir dans le Tableau 5.5 que les deux configurations respectent les requis en matière de fiabilité, déterminisme et latence maximale, malgré l'unicité du bus dans la configuration B. Bien que la majorité des spécifications soient remplies, cette dernière différence induit une charge maximale excédant les spécifications, bien qu'une réduction importante des ressources matérielles puisse également être observée.

5.2.4 Discussion

Pour faire suite à la présentation du processus de design et validation à l'aide de l'article du chapitre 4, la présente discussion portera sur deux thèmes de contribution. La première partie reviendra sur les avantages et limitations des différents types d'analyse incluses dans notre approche mixte de design et validation, tandis que la seconde portera sur les modifications proposées au niveau du protocole ARINC 825 qui viendront appuyer ces mêmes analyses.

5.2.4.1 Plateforme mixte de design et validation

Tel que décrit dans la section précédente, l'approche mixte reposant sur une plateforme matérielle permet la validation complète d'une architecture de réseau avionique en ayant toutefois recours à certaines approches logicielles afin de valider des contraintes de fiabilité. Nous privilégions donc l'emploi d'une plateforme matérielle dans tout le processus de design et validation de systèmes avioniques, ce qui entraîne à notre avis les avantages suivants. Malgré une

hausse initiale de l'effort lors du premier projet suivant cette méthode, la réutilisation, due en partie à l'adoption d'une architecture générique, permet d'abord plusieurs gains substantiels en efficacité [96]. En misant sur la réutilisation, une fonction utilisée dans plusieurs domaines d'applications sur un même aéronef permet un développement unique. Avec l'utilisation de composants aux résultats prévisibles, le nombre d'aspects nouveaux et inconnus s'en trouve considérablement réduit. Ceci peut permettre une réduction de la durée des cycles de design et validation. En se fondant sur ce même principe, une réduction similaire a pu être observée dans le domaine automobile avec l'introduction dans les années 80 de l'ECU [97], proche parent des LRU de l'approche IMA. Ceci pourrait s'avérer particulièrement bénéfique, puisque le temps de développement de systèmes avioniques demeure à ce jour particulièrement long étant donné la quantité importante de normes de toutes sortes auxquelles le système doit se conformer, telles celles présentées au chapitre 2. En nous basant sur la littérature exposée, nous pensons également être les premiers à promouvoir plus explicitement la réutilisation dans le processus de design de systèmes avioniques. Cette réutilisation, centrale à notre approche, permet entre autres de maximiser les avantages découlant des efforts initiaux requis pour l'application des DO-178C ou DO-254.

La compatibilité inhérente de la certification dans l'approche de développement proposée constitue la seconde source de bénéfices. Contrairement à l'approche logicielle qui n'est pas basée sur les applications finales, la certification des composants ne peut être effectuée à partir de cette étape importante du processus. Les tests utilisés dans la validation de notre architecture peuvent également servir lors de la certification, puisque le même composant sera utilisé dans l'implémentation finale. Compte tenu de la littérature exposée au chapitre 2 à ce sujet, nous considérons être les premiers à proposer une approche directement compatible avec les activités de certification. En effet, une proportion importante des travaux exposés au chapitre 2 se limite à souligner l'importance du processus de certification, contrairement à notre approche, qui suit l'ensemble des caractéristiques recommandées pour faciliter la certification. Notre prétention de compatibilité avec les exigences de certification est principalement basée sur les conclusions de l'article [91] énonçant 5 caractéristiques à respecter pour réduire les efforts de certification. Ces 5 caractéristiques ont été énoncées à la section 5.2.2.

La possibilité offerte par une implémentation en temps réel de l'application finale demeure un dernier avantage de l'utilisation d'une plateforme matérielle. Dans le cas où une approche

propriétaire est privilégiée comme dans le cadre de notre projet, une compréhension plus complète des différentes normes peut permettre de détecter des modes d'erreur difficiles à cerner à l'aide de modèles mathématiques. C'est le cas, entre autres, de la modification au protocole de gestion d'erreurs et de l'inclusion de la stratégie de regroupement de trames présentée dans la section suivante. Dans un premier cas, le respect des contraintes de fiabilité a pu être validé, tandis que les performances globales du réseau secondaire ont pu être améliorées dans le second cas. L'ajout d'un système de monitorage souligne, par contre, une limite de l'approche matérielle. En effet, certains processus requièrent finalement moins d'effort lorsqu'implémentés en logiciel plutôt qu'en matériel. Comme le système de monitorage implémenté induit une augmentation de la consommation de la bande passante, une approche logicielle aussi performante aurait bien pu convenir pour un effort similaire sans induire d'effet négatif. L'exposition de tous ces avantages et limites nous amène donc à être les premiers à privilégier une approche mixte pour le design et la validation de systèmes avioniques sous une architecture respectant les principes de l'IMA.

5.2.4.2 Stratégie de regroupement de trames

Afin de maximiser les avantages de l'architecture hiérarchique présentée dans le chapitre 3, nous avons évalué l'efficacité de plusieurs stratégies de regroupement de trames au niveau du bus secondaire respectant la norme ARINC 825. Les stratégies retenues ont pour but de réduire la charge sur les bus ARINC 825 en regroupant l'envoi de trames de plusieurs capteurs dans une seule trame. En combinaison avec l'approche de gestion de la redondance présentée dans la section suivante, les stratégies implémentées permettent d'augmenter encore plus le nombre de capteurs desservis pour une même consommation de ressources. Ce dernier point fera l'objet du reste de cette section en détaillant les principes de fonctionnement des stratégies implémentées. Bien que la démonstration suivante s'appuie sur les avantages induits par l'adoption d'une stratégie spécifique, d'autres stratégies pourraient également convenir. L'efficacité de chaque méthode en fonction de chaque trafic précis peut, par contre, dépendre de la stratégie sélectionnée.

La Figure 5-7 présente la composition de la charge utile de 16 bits requise pour la transmission d'une trame de base générée pour chaque capteur. Cette trame est employée à travers le réseau de capteurs développé et validé à l'aide de la plateforme présentée dans le chapitre 4. À la figure

suivante, les 6 premiers bits contiennent les adresses de la source et de la destination tandis que les 10 bits restants contiennent la valeur du capteur encodé sur ces 10 bits.

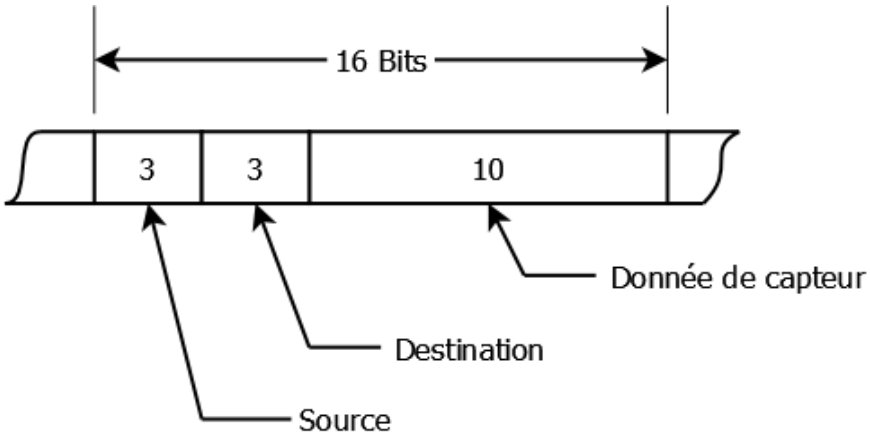


Figure 5-7: Composition de la trame

Nous pouvons donc considérer l'inclusion de la stratégie *M-to-1* dans notre implémentation du contrôleur ARINC 825 comme une amélioration proposée à la norme de base. Une amélioration comme celle-ci est, par contre, permise dans les limites imposées par la norme. Ce type de contrôleur serait toujours conforme à la norme ARINC 825. Selon les travaux recensés à la section 2.3, nous n'avons trouvé aucune mention de travaux suggérant l'introduction de stratégies de regroupement de trames directement dans le contrôleur.

À travers notre implémentation, la valeur de *M* peut être variée pour évaluer l'efficacité pour chacune des valeurs de 1 à 4. La norme ARINC 825 dicte que chaque trame doit contenir 58 bits obligatoires pour le contrôle, la validation et l'adressage, et une charge utile de 64 bits supplémentaires. Le nombre total de bits peut varier légèrement en fonction du remplissage de bit utilisé par les mécanismes de détection et gestion des pannes. Pour un groupe de 4 capteurs, le Tableau 5.6 présente la comparaison des différents trafics obtenus en suivant la stratégie *M-to-1* pour plusieurs valeurs de *M* à l'aide de notre plateforme de validation. La possibilité offerte par une implémentation personnalisée plutôt qu'une approche propriétaire permet, entre autres, d'implémenter ce genre de mécanismes non spécifiés dans la norme, mais toujours dans ses limites. Les résultats du Tableau 5.6 ont été générés en variant M dans le générateur de trames de notre contrôleur ARINC 825.

Les mesures de latence, obtenues à l'aide du système de monitorage mentionné dans [10], ont été conduites sur des intervalles de transmission de 1 sec correspondant à 500 cycles de génération et

transmission de trames pour chaque capteur. Les caractéristiques du bus secondaire obtenues à l'aide de la méthodologie présentée dans la section précédente pour les différentes valeurs de M sont présentées au Tableau 5.6.

Tableau 5.6: Trafic sous la stratégie de regroupement de trames *M-to-1*

M	Latence (ms)		Charge du réseau	Nombre maximal de capteurs
	Moyenne	Maximum		
1	0.24	0.36	36%	5
2	0.19	0.24	24%	8
4	0.20	0.20	20%	12

On peut voir dans le Tableau 5.6 que les valeurs de latence maximale et moyenne observées sur le bus décroissent à mesure que les différentes trames des capteurs sont groupées dans une trame ARINC 825. La réduction dans la latence maximale est due à la réduction des surcoûts de communication inhérents à la transmission à l'aide d'un bus ARINC 825. À travers l'équation suivante correspondant à la manière typique de calculer la charge d'un bus 825, on peut vérifier jusqu'à quel point la réduction du surcoût peut induire une réduction de la charge du réseau.

$$\mathrm{Ld} = \frac{\sum_{n=0}^{K} F_n}{I \bullet BW} \qquad (5.11)$$

où Ld est la charge, K le nombre de trames envoyées sur l'intervalle de transmission sélectionné, Fn la longueur de la trame correspondante, I la durée de l'intervalle de transmission et BW la bande passante totale disponible. Puisque la charge du réseau détermine le nombre maximal de capteurs regroupés dans chaque TIM, le regroupement de trames permet l'inclusion de plus de capteurs réduisant le recours aux bus parallèles. Le nombre maximal de capteurs sur un bus est, par contre, toujours limité par la bande passante maximale offerte. Un nombre plus élevé de capteurs par TIM permet également une meilleure gestion de la redondance en offrant plus de possibilités de regroupement des capteurs non redondants. Cet aspect de la gestion de la redondance, présenté dans le deuxième article de cette thèse, et sera détaillé à la section suivante.

Il est à noter que les stratégies de regroupement de trames sont habituellement implémentées dans un « gateway » qui relie des mécanismes de communication implémentés selon au moins deux protocoles distincts [57]. Notre proposition se démarque ici aussi en ne ciblant qu'un seul protocole pouvant ainsi être utile pour toutes les utilisations de la norme, même si celle-ci est utilisée de manière indépendante.

Au niveau de la modification apportée aux mécanismes de gestion d'erreurs des contrôleurs ARINC 825 énoncés dans la section précédente, ces changements causent cependant le non-respect de la norme. En effet, de manière à pouvoir respecter les contraintes temporelles imposées, on peut voir à la section précédente que l'état de notre contrôleur implémenté change lorsqu'une panne est détectée plutôt que de laisser les compteurs appropriés s'incrémenter graduellement jusqu'à un seuil où l'état du contrôleur est finalement modifié pour indiquer les pannes en cours. Bien que pour contourner ce problème, notre implémentation change d'état à la première détection d'une panne, d'autres solutions auraient pu être envisagées telle l'augmentation de l'incrément des compteurs de gestion d'erreurs. Comme chaque contrôleur est responsable de la gestion de son propre état et ne nécessite pas l'état des autres contrôleurs connectés au même bus, la compatibilité avec un contrôleur 825 standard est toujours assurée malgré le non-respect de la norme causé pour le bon fonctionnement du système de gestion de la redondance du système globale. Cette compatibilité entre notre contrôleur ARINC 825 et un contrôleur standard a été testée positivement à l'aide de l'outil ADS2, permettant l'analyse de trafic sur un bus ARINC 825. Nous n'avons répertorié aucun travail mentionnant une proposition directe d'amendement de la norme pour que l'application de celle-ci cesse de causer une dégradation de performance ou une impossibilité de rencontrer les exigences tel le cas de figure explorée dans cette thèse.

5.3 Gestion modulaire de la redondance

5.3.1 Sommaire

L'adoption de l'architecture IMA, présentement en utilisation dans les systèmes avioniques récents, apporte de nombreuses nouvelles possibilités et défis par rapport à l'approche fédérée. À travers nos travaux de validation d'un réseau de transducteurs basée sur une interprétation architecturale des concepts IMA, nous avons identifié plusieurs aspects du processus de design

nécessitant un approfondissement afin de mieux cerner les différents points à améliorer par rapport aux approches traditionnelles. Au niveau de l'atteinte des cibles de fiabilité, les systèmes avioniques ont toujours majoritairement recours à la redondance pour en assurer le respect. L'article du chapitre 4 [9] présente notre approche de gestion de la redondance spécifique au domaine avionique permettant, entre autres, la sélection du niveau approprié de redondance de manière spécifique pour chacune des couches traversées. Notre approche est illustrée à travers le design et la validation d'un réseau de capteurs basé sur un réseau AFDX principal, supporté par plusieurs bus secondaires suivant la norme ARINC 825. Afin de recréer fidèlement les contraintes du domaine avionique, les points communs de défaillance ou couplages ont été inclus dans la modélisation de notre architecture et dans celle des approches de gestion de redondance classiques. Une première analyse vient démonter un gain de fiabilité substantiel face à l'approche de type « composant » ainsi qu'une fiabilité similaire à l'approche de type « système ». Une deuxième phase vient par contre démontrer la meilleure efficacité au niveau des ressources face à l'approche « système ». À l'aide de dernières analyses, la sensibilité du réseau face à un composant ou aux points de couplage vient faciliter la phase de design en permettant la sélection d'une configuration plus efficace par rapport aux contraintes spécifiées. L'architecture proposée prend le meilleur de chacune des méthodes classiques en éliminant les points communs de défaillance.

5.3.2 Discussion

L'analyse de l'article du chapitre 4 montre bien en premier lieu les limites des approches traditionnelles. L'architecture système permet effectivement l'atteinte des cibles de fiabilité en ne rajoutant pas de points communs de défaillance. Son application systématique à toutes les couches du réseau ne permet, par contre, aucune optimisation au niveau des ressources consommées. Ce type de redondance est présent dans de nombreux travaux [1, 85] inclus dans notre revue de littérature sur le sujet. Les réseaux semblables à ceux du A380 reposent effectivement sur ce type d'approche avec 2 réseaux AFDX indépendants. Notre analyse montre par contre qu'un seul réseau serait surement suffisant pour desservir adéquatement plusieurs systèmes. Un système avec un seul réseau de commutateurs AFDX devrait, par contre, toujours avoir une topologie assurant deux chemins possibles de transmission entre chaque combinaison de source et destination. Une simple topologie circulaire pourrait amplement convenir à satisfaire

cette dernière exigence. Tel que démontré dans le chapitre 4, notre proposition architecturale permet, entre autres, de moduler le niveau de redondance en tenant compte des caractéristiques intrinsèques des différents protocoles en utilisation. Au cours des prochaines années, ces mêmes systèmes seront surement amenés à être mis à jour avec les protocoles de prochaine génération. Le caractère générique de l'architecture hiérarchique du chapitre 4 permet donc à nouveau de concorder avec les besoins du développement IMA en permettant une modification ou une évolution plus aisée.

Le second type de redondance inclus dans notre analyse suit une approche au niveau composant. Comme nous l'avons démontré dans l'article précédent, l'inconvénient majeur de cette approche est l'introduction de points communs de défaillances malgré la possibilité de modulation de la redondance à chaque couche. La fiabilité des systèmes de votation nécessaires à chaque couche devient critique, puisqu'en cas de panne majeure, la couche entière devient non-fonctionnelle malgré les efforts de redondance. Il est à noter que l'architecture COM-MON [6], couramment utilisée dans le domaine avionique, rajoute un système de votation supplémentaire pour chaque canal de monitorage multipliant ainsi le nombre d'apparitions de points communs de défaillance. Bien qu'elles n'aient pas été incluses dans l'article du chapitre 4, les performances de cette approche auraient donc été semblables à l'approche par composant avec une sensibilité encore plus grande des systèmes de votation. Notre architecture permet donc de limiter le nombre de systèmes de votation à 1, tout en modulant le niveau de redondance à chaque couche. Les flots de données provenant de capteurs redondants, doivent par contre, pouvoir être distribués de manière indépendante à travers le réseau. Tout réseau doit ici aussi assurer minimalement une connexion indépendante en mode de communication normal avant toute panne. Ici encore, une topologie circulaire permettrait de respecter les exigences demandées. De manière globale, la gestion de redondance incluse dans notre architecture hiérarchique jumelle les avantages des deux techniques traditionnelles en permettant la mitigation de leurs inconvénients.

CHAPITRE 6 CONCLUSION

Ce chapitre résume les travaux exposés jusqu'à présent dans cette thèse. Les diverses limitations rencontrées et les améliorations possibles des travaux effectués ainsi que les possibilités de travaux futurs et pistes de recherche seront, par la suite, proposées.

6.1 Synthèse

Tout au long de cette thèse, nous avons recensé de nombreux travaux de recherche dans le domaine global du design et de la validation des systèmes avioniques. De manière plus précise, nos travaux ont porté plus spécifiquement sur l'introduction de la norme IEEE 1451 en avionique, du développement d'une plateforme matérielle de design et validation, ainsi que sur un nouveau mode de gestion de la redondance dans ce type de système. Dans le contexte de développement de système avionique suivant la norme IMA, nous avons tenté à travers cette thèse d'identifier des lacunes spécifiques dans le processus de design et de validation actuel et de proposer des améliorations possibles afin d'obtenir un processus plus adapté à l'emploi du paradigme IMA.

Dans une optique de faciliter l'intégration d'un nombre grandissant de capteurs requis dans les systèmes de prochaine génération, une architecture hiérarchique de réseaux de capteurs s'appuyant sur le modèle de base de la norme IEEE 1451 a été proposée dans le cadre de l'article [8] du chapitre 3. L'utilisation d'une plateforme matérielle reconfigurable nous a permis l'ajout de mécanismes supplémentaires conférant le niveau de fiabilité requis pour une application critique, ainsi qu'un caractère réutilisable et une meilleure consommation de ressources face à aux architectures similaires. Selon les contraintes à respecter, une architecture sélectionnée permet la validation de la bande passante offerte, du niveau de fiabilité recherché ainsi que du type et de la quantité de transducteurs utilisés. Notre architecture permet également des gains raisonnables au niveau de la consommation de ressources face à des architectures traditionnelles. Dans le cadre d'applications critiques, les écarts de consommation de ressources observés démontrent encore une fois la supériorité de notre architecture.

Dans un second temps, une évolution constante des technologies de communication demande un processus de design et de validation ayant la souplesse d'intégrer efficacement tout nouveau protocole ou périphérique. Nous avons donc présenté à la section 5.2 une approche de design et

de validation de systèmes avioniques à travers une plateforme de prototypage pour de tels systèmes et une architecture générique. Les contraintes de performance et de fiabilité à respecter peuvent être validées, incluant même l'injection de pannes et la mesure de la latence des trames parcourant le réseau. La dernière partie de la section 5.2 présente un exemple de développement d'un réseau de transducteurs basé sur les réseaux ARINC 825 et AFDX. À l'aide d'une meilleure connaissance des différents protocoles et une approche structurée, les processus de design, validation, intégration, certification et entretien de systèmes avioniques complexes et critiques peuvent grandement en bénéficier.

Dans la revue de littérature présentée au chapitre 2, la modélisation logicielle domine les processus de design actuels en permettant la précision requise, malgré un effort parfois important de modélisation. Bien que les limites de l'approche matérielle ne permettent pas encore de garantir certaines contraintes, tels certains aspects de la fiabilité, son utilisation en conjonction avec les modèles logiciels permet plusieurs avantages marquants. Le temps de design se retrouve réduit en ayant recours à une plateforme reconfigurable et générique, en particulier pour satisfaire les contraintes de certification. L'emploi d'une plateforme matérielle peut également amener plusieurs autres avantages, tels le test en temps réel et le support à la certification. Une approche propriétaire permet également de détecter et corriger plus efficacement certains comportements erronés du système, difficiles à détecter et intégrer à tout processus de modélisation. À notre avis, le recours aux deux types d'approches, logicielle et matérielle, est donc nécessaire afin de faciliter le design et la validation de réseaux avioniques suivant la norme IMA.

La dernière contribution contenue dans cette thèse se situe au niveau de la gestion typique de la redondance dans les systèmes avioniques, soit l'introduction de redondance. Bien que la redondance demeure incontournable afin de satisfaire les exigences élevées de fiabilité, nous avons identifié le processus d'introduction de la redondance comme pouvant bénéficier d'une nouvelle approche spécifiquement conçue pour les applications avioniques. L'article du chapitre 4 [9] vient donc présenter notre approche de gestion de la redondance permettant, entre autres, la sélection du niveau approprié de redondance de manière spécifique pour chacune des couches traversées. En prenant soin d'identifier et d'inclure dans nos modèles les points communs de défaillance, nos analyses permettent de conclure que l'approche proposée atteint un niveau de fiabilité égal ou supérieur aux techniques traditionnelles présentement utilisées. Notre approche permet également d'optimiser la consommation de ressources face à ces mêmes approches.

L'architecture proposée prend le meilleur de chacune des méthodes classiques en éliminant les points communs de défaillance. La prochaine section vient présenter les limitations des présents travaux ainsi les possibles voies de continuation.

6.2 Limitations et travaux futurs

La première limitation majeure des travaux présentés dans le cadre de l'intégration de la norme IEEE 1451 découle de l'effort extrêmement important pour une implémentation complète. Dans une optique de réutilisation d'une plateforme matérielle pouvant accommoder de nouveaux types de transducteurs, l'effort de développement requis décroit pour chaque projet subséquent basé sur la même architecture. Ceci explique en majeure partie pourquoi nous n'avons implémenté qu'une partie des processus et directives inclus dans la norme IEEE 1451. Cette approche est même recommandée par la NASA pour une première implémentation pour ajouter par la suite de nouvelles fonctionnalités au fur et à mesure d'utilisations subséquentes. Nous nous sommes donc limités à l'implémentation des fonctions indispensables au contrôle de capteurs et l'acquisition de données. Les modes de contrôle des actuateurs ne sont, par exemple, pas implémentés, sauf pour ceux partagés avec le contrôle des capteurs. Cette implémentation partielle ouvre donc la porte vers une implémentation future plus complète afin d'encore mieux prédire les gains liés à la nouvelle architecture proposée.

Au niveau de la plateforme matérielle de design et de validation de systèmes avioniques, le nombre limité de cartes et de capteurs physiques a entraîné une multiplication des cas de test. Afin de pouvoir émuler un système plus complexe, les transmissions de nombreux capteurs ont dû être générées artificiellement pour modéliser les différents cas de figure voulus. L'augmentation des ressources matérielles de la plateforme permettrait la réduction de l'effort de développement des cas de figure pertinents en permettant de les effectuer de manière directe sur la plateforme. Une plateforme plus complexe n'aurait pas une incidence marquée sur la phase d'implémentation comme telle, puisque les mêmes modules génériques devraient être implémentés. Une prochaine étape pourrait être l'automatisation des cas de test en fonction des critères à valider et de la configuration architecturale sélectionnée. Un lien automatisé pourrait également venir relier l'architecture sélectionnée et les analyses de fiabilité. Pour l'instant, les analyses de fiabilité effectuées à l'aide de Matlab viennent dicter l'architecture à implémenter. L'analyse des cas de test en combinaison avec le matériel disponible vient, par la suite, dicter les

composants à programmer ainsi que les tests à implémenter. Un outil complet partant de l'optimisation de l'architecture en fonction des contraintes jusqu'à la génération des tests et des configurations de la plateforme matérielle pourrait venir augmenter l'efficacité de la méthode et ainsi pouvoir la tester dans plusieurs autres cas de figures et domaines d'applications. Puisque les travaux présentés ont été effectués dans le cadre du projet AVIO 402 ne portant que sur un réseau en particulier, l'ajout d'autres applications viendrait surement renforcer les conclusions énoncées dans cette thèse.

Finalement, au niveau de la proposition de gestion de la redondance, une partie de théorie sur la redondance n'a pas été incluse ni même considérée dans les analyses présentées. En effet, l'ajout de composants redondant inactif en attente peut traditionnellement venir augmenter la fiabilité des systèmes considérés. Comme pour l'approche de type composant, il serait possible et intéressant de voir l'effet des points de défaillance communs sur cette approche. Comme notre approche de gestion de la redondance a été effectuée dans le cadre du projet AVIO 402 ne ciblant qu'une application précise, l'analyse d'autres applications dans des domaines connexes impliquant possiblement des variations des niveaux de redondance pour différentes parties du réseau serait également pertinente afin de venir appuyer les prétentions contenues dans cette thèse.

RÉFÉRENCES

[1] I. Moir et al., *Aerospace Series : Civil Avionics Systems (2nd Edition)*. Somerset, NJ, USA: John Wiley & Sons, 2013.

[2] ARINC, 2009, ARINC Specification 429, Annapolis, Maryland.

[3] ARINC, 2009, Aircraft Data Network Part7, Avionic Full Duplex Switched Ethernet, Annapolis, Maryland.

[4] ARINC, 2011, ARINC Specification 825-2, Annapolis, Maryland.

[5] Bosch, 1991, "CAN Specification Version 2.0."

[6] S. Bouanen, "Interface de transducteurs intelligents tolerante aux pannes pour des applications avioniques critiques," ProQuest Dissertations Publishing, 2014.

[7] "IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats," p. 1-335, 2007.

[8] J.-P. Tremblay et al., "A System Architecture for Smart Sensors Integration in Avionics Applications," SAE International Journal of Aerospace, vol. 5, n°. 1, p. 7, October 2012.

[9] J. P. Tremblay et al., "Hierarchical Redundancy Management for Avionic Networks," IEEE Aerospace and Electronic Systems Magazine, 2015.

[10] J. P. Tremblay et al. A hardware prototype for integration, test and validation of avionic networks, IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC), pp. 2D5-1-2D5-11, 2013.

[11] A. Kumar et al., "Current Status of the IEEE 1451 Standard-Based Sensor Applications," IEEE Sensors Journal, vol. 15, n°. 5, p. 2505-2513, 2015.

[12] M. Grisostomi et al. Modular design of a novel wireless sensor node for smart environments, IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA), 1-5, 2014.

[13] IEEE, "IEEE Standard for Local and metropolitan area networks--Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)," p. 1-314, 2011.

[14] B. Nikola et P. Ivan, "Service-oriented implementation model for smart transducers network," Computer Standards and Interfaces, vol. 38, p. 78-83, February 2015 2015.

[15] R. Alena et al., "Wireless Space Plug-and-Play Architecture (SPA-Z)", IEEE.Aerospace Conference, p. 1-17, 2014.

[16] B. Federico et al., "Networked transducers in intelligent transportation systems based on the IEEE 1451 standard," Computer Standards and Interfaces, vol. 36, no. 2, p. 11, February 2014.

[17] J. Guevara et al. "Open architecture for WSN based on runtime reconfigurable systems and the IEEE 1451", IEEE SENSORS, p. 1-4 ,2013.

[18] J. Xin et al., "Fault cases management system for avionics system with the characteristics of network", 2014 International Conference on Reliability, Maintainability and Safety (ICRMS), p .826-830, August 2014.

[19] S. Trujillo et al., "MultiPARTES: Multicore Virtualization for Mixed-Criticality Systems", 2013 Euromicro Conference on Digital System Design (DSD), p. 260-265, September 2013.

[20] D. Z. Lu et H. Yang, "Unlocking the Power of OPNET Modeler", Cambridge University Press, 2012.

[21] W. Haotian et X. Huagang., "A novel data communication network architecture for integrated modular avionics", IEEE/AIAA 28th Digital Avionics Systems Conference, p. 7.B.1-1-7.B.1-8, October 2009.

[22] C. Zhao et al., "Real-time performance evaluation of avionics networks using network calculus," Journal of Information and Computational Science , vol. 12, n°. 9, p. 8, 10 Juin 2015.

[23] N. Safwat et al., " Avionics Full-duplex switched Ethernet (AFDX): Modeling and simulation", 32nd National Radio Science Conference (NRSC), 2015.

[24] A. Louati et al., "Temporal properties verification of real-time systems using UML/MARTE/OCL-RT," Advances in Intelligent Systems and Computing , vol. 346, p. 14, 2015.

[25] M. Li et al., "Incorporating Performance Analysis into Reliability Assessment for Avionics Full-Duplex Switched Ethernet Networks," Reliability Engineering & System Safety, 2016.

[26] M. Li et al., "Reliability Enhancement of Redundancy Management in AFDX Networks," IEEE Transactions on Industrial Informatics, 2016.

[27] L. Huan et al. "A systematic approach for safety evidence collection in the safety-critical domain", 9th Annual IEEE International Systems Conference (SysCon), p. 194-199, April 2015.

[28] A. Garro et A. Tundis, "On the Reliability Analysis of Systems and SoS: The RAMSAS Method and Related Extensions," IEEE Systems Journal , vol. 9, n°. 1, p. 232-241, 2015.

[29] A. Garro et al., "System reliability analysis: A model-based approach and a case study in the avionics industry," Proceeding of 3rd CEAS, p. 12, 2011.

[30] A.-R. Guduvan et al., "Test languages for in-the-loop avionics tests," Journal of Aerospace Information Systems , vol. 12, p. 17, 2015.

[31] TechSat, "*ADS2 – Avionic Development System*," version.

http://www.techsat.com/products/software/ads2.html, 2015.

[32] G. Afonso et al., "Heterogeneous CPU/FPGA Reconfigurable Computing System for Avionic Test Application", IEEE 27th International Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), p. 260-267 , Mai 2013.

[33] Z. Hao et al. Design and implementation of DMT-based high-speed avionics bus interface, 8th International Conference on Computer Science & Education (ICCSE), p. 664-667, April 2013.

[34] A. Ben Achballah et S. Ben Saoud., "The design of a Network-On-Chip architecture based on an avionic protocol", World Symposium on Computer Applications & Research (WSCAR), January 2014.

[35] D. Trentin et al., "An AFDX Switch Fabric Hardware Core for Avionic Network Prototyping and Characterization," SAE International Journal of Aerospace, vol. 5, n°. 1, p. 7, October 2012.

[36] R. Beneder et al., " Runtime verification infrastructure for Embedded Linux", IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA), 2014.

[37] S. Esquembri et al.," Hardware timestamping for image acquisition system based on FlexRIO and IEEE 1588 v2 standard", 19th IEEE-NPSS Real Time Conference (RT), 2014.

[38] T. Schuster et D. Verma., "Networking concepts comparison for avionics architecture", IEEE/AIAA 27th Digital Avionics Systems Conference, 2008.

[39] L. Renjun et al., "A design for automotive CAN bus monitoring system", Vehicle Power and Propulsion Conference, 2008.

[40] B. Liu et T. Ji., "The design of monitoring system based on CAN bus", International Conference on Measurement, Information and Control (MIC), 2012.

[41] T. Ziermann et al., "FPGA-based testbed for timing behavior evaluation of the Controller Area Network (CAN)", International Conference on Reconfigurable Computing and FPGAs (ReConFig), 2012.

[42] O. Acevedo et al., "Towards optimal design of avionics networking infrastructures", IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), 2012.

[43] H. Kimm et M. Jarrell., "Controller Area Network for fault tolerant small satellite system design", IEEE 23rd International Symposium on Industrial Electronics (ISIE), 2014.

[44] S. Kelkar et R. Kamal., "Implementation of data reduction technique in Adaptive Fault Diagnosis Algorithm for Controller Area Network", International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), 2014.

[45] I. Sheikh et al.," Improving information throughput and transmission predictability in Controller Area Networks", IEEE International Symposium on Industrial Electronics (ISIE), 2010.

[46] X.-D. Han et al., "Design of dual redundancy CAN-bus controller based on FPGA", 8th IEEE Conference on Industrial Electronics and Applications (ICIEA), 2013.

[47] K. Szurman et al., "Fault tolerant CAN bus control system implemented into FPGA", IEEE 16th International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS), 2013.

[48] B. Groza et S. Murvay, "Efficient Protocols for Secure Broadcast in Controller Area Networks," IEEE Transactions on Industrial Informatics, vol. 9, n°. 4, p. 2034-2042, 2013.

[49] K. Sandstrom et al., "Frame packing in real-time communication", Seventh International Conference on Real-Time Computing Systems and Applications, 2000.

[50] K. Minkoo et al., "Frame Packing for Minimizing the Bandwidth Consumption of the FlexRay Static Segment," IEEE Transactions on Industrial Electronics vol. 60, no. 9, p. 4001-4008, 2013.

[51] L. Feng et al., "Research on FlexRay communication system", Vehicle Power and Propulsion Conference, 2008.

[52] N. Abbas et al., "Optimal WiMAX frame packing for minimum energy consumption", 7th International Wireless Communications and Mobile Computing Conference (IWCMC), 2011.

[53] IEEE, "IEEE Standard for WirelessMAN-Advanced Air Interface for Broadband Wireless Access Systems," p. 1-1090, 2012.

[54] S. Misbahuddin et al., "An efficient lossless data reduction algorithm for cluster based wireless sensor network", International Conference on Collaboration Technologies and Systems (CTS), 2014.

[55] B. Tanasa et al., "Reliability-aware frame packing for the static segment of flexray," Proceedings of the ninth ACM international conference on Embedded software, Taiwan, p. 175-184, 2011.

[56] L. Meng et al., "Determinism Enhancement of AFDX Networks via Frame Insertion and Sub-Virtual Link Aggregation," IEEE Transactions on Industrial Informatics , vol. 10, no. 3, p. 1684-1695, 2014.

[57] H. Ayed, "Analyse et Optimisation des réseaux avioniques hétérogènes," Institut National Polytechnique de Toulouse, 2014.

[58] A. Hemalatha et R. Venkatesh., "Redundancy management in heterogeneous wireless sensor networks", International Conference on Communications and Signal Processing (ICCSP), 2014.

[59] F. Baronti et al., "Mitigation of Single Event Upsets in the control logic of a charge equalizer for Li-ion batteries", 39th Annual Conference of the IEEE Industrial Electronics Society, IECON 2013.

[60] M. Amiri et V. Prenosil., "Significant reliability improvement of NMR systems, International Conference on Military Technologies (ICMT), 2015.

[61] A. Kwiecień et J. Stój, "The Cost of Redundancy in Distributed Real-Time Systems in Steady State," dans *Computer Networks*, vol. 79, A. Kwiecień, P. Gaj, et P. Stera, Édit.: Springer Berlin Heidelberg, p. 106-120, 2010.

[62] S. Askari et M. Nourani, "Design methodology for mitigating transient errors in analogue and mixed-signal circuits," Circuits, Devices & Systems, vol. 6, n°. 6, p. 447-456, 2012.

[63] L. Sterpone et A. Ullah., "On the optimal reconfiguration times for TMR circuits on SRAM based FPGAs", NASA/ESA Conference on Adaptive Hardware and Systems (AHS), 2013.

[64] R. Santos et al., "Dynamically Adaptive Scrubbing Mechanism for Improved Reliability in Reconfigurable Embedded Systems," Proceedings of the 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC), p. 6, 2015.

[65] B. Vibishna et al., "Understanding single-event effects in FPGA for Avionic system design," IETE Technical Review, vol. 30, n°. 6, p. 497-505, Nov-Dec 2013.

[66] M. P. Baze et al., "A digital CMOS design technique for SEU hardening," IEEE Transactions on Nuclear Science, vol. 47, n°. 6, p. 2603-2608, 2000.

[67] E. H. Cannon et al., "SRAM SER in 90, 130 and 180 nm bulk and SOI technologies", IEEE International Reliability Physics Symposium Proceedings, 2004.

[68] R. L. Alena et al., "Communications for Integrated Modular Avionics, IEEE Aerospace Conference, 2007.

[69] R. Black et M. Fletcher, "Next generation space avionics: layered system implementation," IEEE Aerospace and Electronic Systems Magazine, vol. 20, no. 12, p. 9-14, 2005.

[70] L. H. Eccles, "The need for smart transducers: an aerospace test and evaluation perspective," IEEE Instrumentation & Measurement Magazine, vol. 11, no. 2, p. 23-28, 2008.

[71] L. Suk et al., "Implementation of dual redundant CAN module based on IEEE 1451 in in-vehicle network", Proceedings of the 2004 IEEE International Conference on Control Applications, 2004.

[72] B. Joshi et al., "Hierarchical plug-and-play self-diagnosable intelligent sensor networks for process control", IEEE International Conference on,Electro/Information Technology, 2008.

[73] M. R. Islam et al., "Energy Efficient Cooperative Technique for IEEE 1451 Based Wireless Sensor Network", Wireless Communications and Networking Conference, 2008.

[74] M. L. Shooman, "Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design", John Wiley & Sons Inc., 2002.

[75] U. D. o. Defense, "Military Handbook - Reliability Prediction of Electronic Equipment - MIL-HDBK-227F," 1991.

[76] C. B. Watkins et R. Walter., "Design considerations for systems hosted on Integrated Modular Avionics platforms", IEEE/AIAA 27th Digital Avionics Systems Conference, 2008.

[77] S. Dajiang et al., "Filling the gap between IMA development and safety assessment through safety-driven model-based system engineering", IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), 2012.

[78] S. Xiaoxuan et K. S. McElvain, "Time Multiplexed Triple Modular Redundancy for Single Event Upset Mitigation," IEEE Transactions on Nuclear Science, vol. 56, no. 4, p. 2443-2448, 2009.

[79] A. Ejlali et al., "Combined time and information redundancy for SEU-tolerance in energy-efficient real-time systems," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 14, no. 4, p. 323-335, 2006.

[80] R. V. Kshirsagar et R. M. Patrikar, "Design of a novel fault-tolerant voter circuit for TMR implementation to improve reliability in digital circuits," Microelectronics Reliability, vol. 49, n°. 12, p. 1573-1577, 12// 2009.

[81] F. Siegle et al., "New voter design enabling hot redundancy for asynchronous network nodes", NASA/ESA Conference on Adaptive Hardware and Systems (AHS), 2014.

[82] S. Gohil et al., "Redundancy management and synchronization in avionics communication products", Navigation and Surveilance Conference (ICNS)Integrated Communications, 2011.

[83] A. Manuzzato et al., "Effectiveness of TMR-based techniques to mitigate alpha-induced SEU accumulation in commercial SRAM-based FPGAs", 9th European Conference on Radiation and Its Effects on Components and Systems, 2007.

[84] M. McCabe et al., "Avionics architecture interface considerations between constellation vehicles", IEEE/AIAA 28th Digital Avionics Systems Conference, 2009.

[85] F. K, "Principles of fly-by-wire architectures," Chalmers University of Technology, Sweden, 2003.

[86] M. S. Haouati, "Architectures innovantes de systèmes de commandes de vol," Institut National Polytechnique de Toulouse (INPT), 2010.

[87] U. D. o. Defense, "Military Handbook – Electronic Reliability Design Handbook - MIL-HDBK-338B," 1998.

[88] T. Chao-Wen et al., "Extending EPCglobal ALE middleware to integrate transducer capability of IEEE 1451 standards", Sixth International Conf on Ubiquitous and Future, 2014.

[89] F. Adamo et al., "A Smart Sensor Network for Sea Water Quality Monitoring," IEEE Sensors Journal, vol. 15, no. 5, p. 2514-2522, 2015.

[90] M. D. R. Perera et al., "A single-chip solution for interfacing transducers to sensor networks using FPGAs", 8th International Conference on Computer Science & Education (ICCSE), 2013.

[91] J. P. Keithan et al., "The Use of Advanced Verification Methods to Address DO-254 Design Assurance", IEEE Aerospace Conference, 2008.

[92] S. Bouanen et al., "Fault tolerant smart transducer interfaces for safety-critical avionics applications", IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC), 2013.

[93] Xilinx, "UG526 SP605 Hardware User Guide," July 18 2011.

[94] Avnet, "ISM Networking FMC Module User Guide," June 29 2010.

[95] R. L. Oostdyk et al., "A Kennedy Space Center implementation of IEEE 1451 networked smart sensors and lessons learned", IEEE Aerospace Conference, 2009.

[96] P. Simpson, "FPGA design: best practices for team-based design", vol. 1st Edition. New York, Springer Verlag, 2010.

[97] J. Munoz-Castaner et al., "A Review of Aeronautical Electronics and Its Parallelism With Automotive Electronics," IEEE Transactions on Industrial Electronics, vol. 58, n^{o}. 7, p. 3090-3100, 2011.