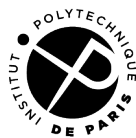


# Ceci est ma thèse

Vie amoureuse des tétards à Ph7

**Maxime Ayrault**

A thesis presented for the degree of  
Doctor of Philosophy



**IP PARIS**

Department Name

Telecom Paris

France

30 février 2056

# **Ceci est ma thèse**

Vie amoureuse des tétards à Ph7

**Maxime Ayrault**

## **Résumé**

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

## **Abstract**

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Dédicace

À ma Maman adorée...

# Remerciements

Je remercie Jean-Do pour ces conseils avisés.....

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>8</b>
<b>2</b>	<b>La sécurité dans le monde de l'automobile</b>	<b>9</b>
2.1	La révolution logicielle . . . . .	9
2.2	Le risque cybersécurité . . . . .	10
<b>3</b>	<b>L'architecture calculateur</b>	<b>12</b>
3.1	Section Title . . . . .	12
<b>4</b>	<b>Expression de la résilience</b>	<b>14</b>
4.1	Section Title . . . . .	14
<b>5</b>	<b>Conclusions et perspectives</b>	<b>15</b>
5.1	Les leçons apprises . . . . .	15
5.2	Les frontières de la recherche . . . . .	15
<b>A</b>	<b>Ma belle annexe</b>	<b>17</b>
A.1	Introduction . . . . .	17

# Table des figures

3.1	Architecture générique . . . . .	13
-----	----------------------------------	----

# Liste des tableaux

2.1	Système logiciel dans une voiture . . . . .	10
4.1	very basic table . . . . .	14

# Liste des abbréviations

- **CAN** : Controller Area Network.
- **ECU** : Electronic Controller Unit.



# Chapitre 1

## Introduction

*Il est très important, pour celui qui souhaite découvrir, de ne pas limiter son esprit à un seul chapitre de la science mais plutôt de rester en contact avec plusieurs autres.*

Jacques Hadamard.

Nous vivons dans l'ère des télécommunications. Les communications constituent, sans aucun doute, l'une des plus importantes révolutions de la science et la technologie.

Ce mémoire de thèse .....

Ce mémoire est divisé en six chapitres et 2 annexes..

Le chapitre 2 contient un rappel sur la théorie....

Le chapitre 3 développe....

Au chapitre 4, expose une application des résultats ....

Le chapitre 5 déjà la conclusion et les perspectives .....

Dans l'espoir de pouvoir intéresser le lecteur curieux, .....

# Chapitre 2

## La sécurité dans le monde de l'automobile

*Ce travail fût présenté en partie à la conférence des sourds-muets unijambistes à Quibéron en avril 1994.*

Maxime Ayrault

### 2.1 La révolution logicielle

Le logiciel embarqué est une des innovations clé dans le monde automobile. D'après l'article de Robert Charette (CHARETTE 2009) paru dans IEEE Spectrum, la première voiture embarquant un logiciel était la Oldsmobile Toronado de General Motors en 1977. La Toronado avait une unité de contrôle électronique (ECU) qui gérant la synchronisation des bougies (spark timing). En 1978, General Motors proposait en option sur ses Cadillacs un ordinateur de bord qui pouvait afficher la vitesse, le niveau du réservoir d'essence, les informations sur l'état du véhicule. Ce logiciel s'exécutait sur une version modifiée du processeur Motorola 6802 et faisait 50.000 lignes de code. Depuis, de plus en plus de fonctions ont été réalisées par du logiciel. Afin de limiter le nombre de câbles dans une voiture, les capteurs ont été décentralisés et connectés à un réseau interne (réseau CAN). Le logiciel a également permis de créer de nouvelles fonctions pour les voitures.

Une voiture actuelle est maintenant devenue une plate-forme logiciel sur roues. Une voiture grand public contient entre 30 et 50 unités de contrôle électronique effectuant la gestion de multiples systèmes (voir 2.1).

Air bag	ABS	Système d'alarme
La climatisation	Le régulateur de vitesse	Le régime moteur
Les clignotants	Les feux	Le klaxon
La gestion des sièges	Le système de navigation	Le système audio
la pression des pneus	les vitres	...

TABLE 2.1 – Système logiciel dans une voiture

En 2009, Alfred Katzenbach, le directeur des Technologies Informatique Chez Daimler, a annoncé que le système navigation et audio sur une Mercedes-Benz S-class contient plus 20 millions de lignes de code et que la voiture contient pratiquement autant d'unité électronique qu'un Airbus A380 (si on exclut le système de divertissement en vol) (CHARETTE 2009). Les logiciels dans une voiture grossissent à une vitesse exponentielle en taille et en complexité. En 2010, certaines voitures avaient 10 millions de lignes de code ; an dix ans, ceci a augmenté par un facteur 15, pour environ 150 millions de lignes. Aujourd'hui, le Model S de Tesla est équipée d'un écran tactile de 17 pouces basé sur un système d'exploitation Linux qui contrôle quasiment toute les fonctions de la voiture, de la performance moteur au système audio et de navigation. En fait, il n'y a plus que 2 boutons manuel ; le bouton pour les feux de détresse et le bouton de la boîte à gants.

Aujourd'hui, le coût du logiciel et de l'électronique représente entre 35 à 40% du coût d'une voiture.

Il n'est pas rare que les manuels utilisateurs des voitures fassent maintenant plus de 500 pages pour expliquer l'ensemble des fonctions logiciel. On estime qu'un conducteur moyen n'utilise pas plus de 20Ceci a aussi d'importantes conséquences sur la façon d'entretenir et de réparer une voiture. On estime que plus de 50

**Todo list 1** *Introduire Automotive Open System Architecture (AUTOSAR). Une solution pour réduire le coût du logiciel en augmentant l'interopérabilité entre les constructeurs.*

L'introduction du logiciel a permis d'avoir des voitures plus sûrs et moins polluantes mais a également introduit une nouvelle menace ; la *sécurité des voitures*.

## 2.2 Le risque cybersécurité

L'ajout de logiciels et de connectivité....

**Todo list 2** <https://www.nytimes.com/2015/09/27/business/complex-car-software-b.html> *Fear of Hacking Andy Greenberg steered a 2014 white Jeep Cherokee down a highway in St. Louis, cruising along at 70 miles per hour. Miles away, two local hackers, Charlie Miller and Chris Valasek, sat on a leather couch at Mr. Miller's house, laptops open, ready to wreak havoc.*

*As Mr. Greenberg sped along, both hands on the wheel, his ride began to go awry. First, the air-conditioning began blasting. Then an image of the hackers in tracksuits appeared on the digital display screen. Rap music began blaring at full volume, and Mr. Greenberg could not adjust the sound. The windshield wipers started and cleaning*

*fluid sprayed, obstructing his view. Finally, the engine quit. Mr. Greenberg was on a highway with no shoulder. A big rig blew past, blaring its horn.*

*"I'm going to pull over," Mr. Greenberg said. "Because I have PTSD."*

*The episode was in fact a stunt orchestrated by the hackers and Mr. Greenberg, a writer for Wired magazine, to demonstrate the Jeep's very real vulnerabilities. The article appeared on July 21.*

*Days later, Fiat Chrysler, the maker of Jeep, announced a recall of 1.4 million vehicles to fix the flaws the hackers had identified - the first known recall intended to address a possible hacking threat.*

# Chapitre 3

## L'architecture calculateur

*Il est très important, pour celui qui souhaite découvrir, de ne pas limiter son esprit á un seul chapitre de la science mais plutôt de rester en contact avec plusieurs autres.*

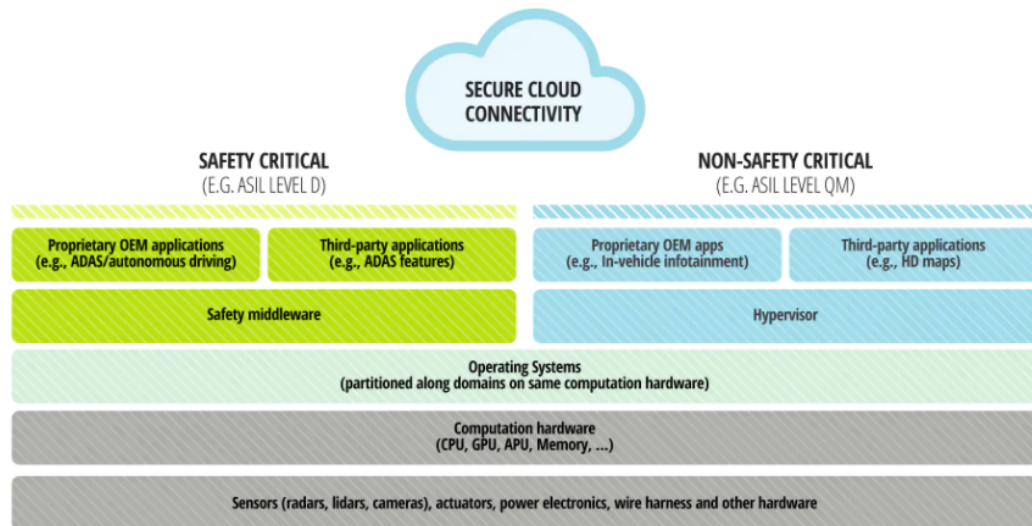
Jacques Hadamard.

### 3.1 Section Title

**Todo list 3** *Commencer á présenter les différentes couches logiciel.*

*the following four stacks could become the basis for upcoming generations of cars in five to ten years :*

- Time-driven stack. *In this domain, the controller is directly connected to a sensor or actuator while the systems have to support hard real-time requirements and low latency times ; resource scheduling is time based. This stack includes systems that reach the highest Automotive Safety Integrity Level classes, such as the classical Automotive Open System Architecture (AUTOSAR) domain.*
- Event- and time-driven stack. *This hybrid stack combines high-performance safety applications, for example, by supporting ADAS and HAD capability. Applications and peripherals are separated by the operating system, while applications are scheduled on a time base. Inside an application, scheduling of resources can be based on time or priority. The operating environment ensures that safety-critical applications run on isolated containers with clear separation from other applications within the car. A current example is adaptive AUTOSAR.*
- Event-driven stack. *This stack centers on the infotainment system, which is not safety critical. The applications are clearly separated from the peripherals, and resources are scheduled using best-effort or event-based scheduling. The stack contains visible and highly used functions that allow the user to interact with the vehicle, such as Android, Automotive Grade Linux, GENIVI, and QNX.*



Source: Deloitte research, Aptiv.

FIGURE 3.1 – Architecture générique

- Cloud-based (off-board) stack. *The final stack covers and coordinates access to car data and functions from outside the car. The stack is responsible for communication, as well as safety and security checks of applications (authentication), and it establishes a defined car interface, including remote diagnostics.*

<https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/rethinking-car-software-and-electronics-architecture>

<https://www2.deloitte.com/us/en/insights/focus/future-of-mobility/pure-play-software.html>

# Chapitre 4

## Expression de la résilience

*Il est très important, pour celui qui souhaite découvrir, de ne pas limiter son esprit à un seul chapitre de la science mais plutôt de rester en contact avec plusieurs autres.*

Jacques Hadamard.

### 4.1 Section Title

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

A	B	C
1	2	3
4	5	6

TABLE 4.1 – very basic table

# Chapitre 5

## Conclusions et perspectives

*Je vois refléter dans mon miroir tout mon  
passé et tout mon avenir.*

J. Cortázar.

### 5.1 Les leçons apprises

Aucune !

### 5.2 Les frontières de la recherche

The sky is the limit...



# Bibliographie

CHARETTE, Robert N. (2009). “This car runs on code”. In : *IEEE Specturm* 7649.

# Annexe A

## Ma belle annexe

### A.1 Introduction

La théorie....

Ceci est une appendice...