

## Systèmes Cyber-Physiques : Introduction Automatique et Traitement du Signal

Marc Pantel  
(d'après Yamine Aït Ameur)  
IRIT/INP-ENSEEIH  
Marc.Pantel@n7.fr

Cours  
2018-2019

## Objectif du cours

- ◆ Etapes de développement de systèmes hybrides
  - ❖ Modélisation mathématique
  - ❖ Conception d'une solution
  - ❖ Validation par simulation
  - ❖ Code embarqué sur le système
- ◆ Contexte des systèmes cyber-physiques
  - ❖ => identifier les problèmes techniques
  - ❖ => identifier les solutions (dans le domaine du traitement de l'information)
  - ❖ => convaincre qu'il s'agit d'un domaine aux enjeux importants !
  - ❖ => convaincre que l'informatique et les techniques de modélisation de l'informatique et des mathématiques sont fondamentales pour cet enjeu !

## Quelques exemples Aéronautique

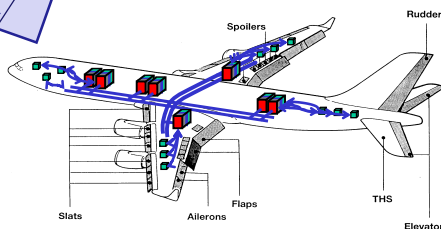


Système

Environnement du système

Aérodynamique

Moteurs  
Surfaces de contrôle



## Quelques exemples Espace



Système

Gravité / Mécanique Classique

Mécanique Relativiste

Moteurs  
Surfaces de contrôle

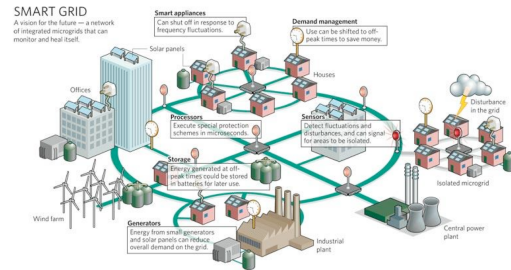
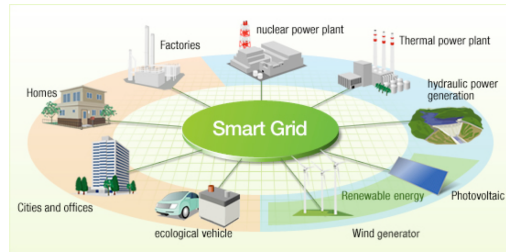
Environnement  
du système

Consigne

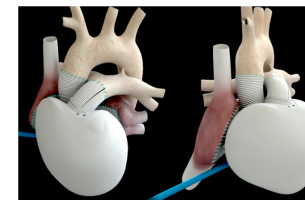
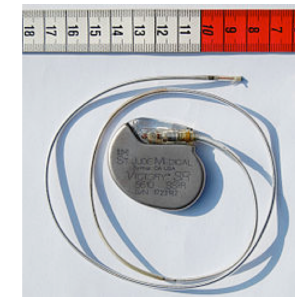
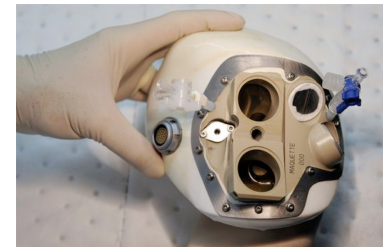
Consigne



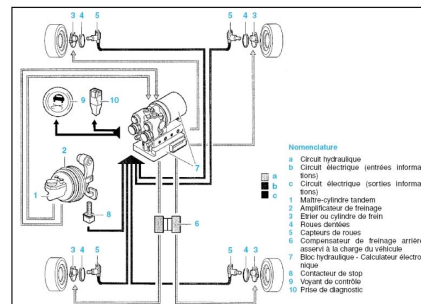
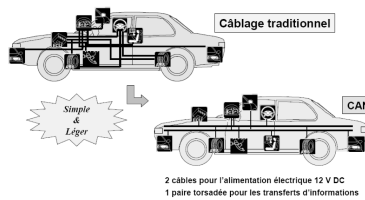
## Quelques exemples Energie



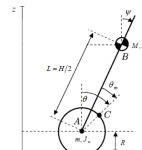
## Quelques exemples Médecine



## Quelques exemples Automobile

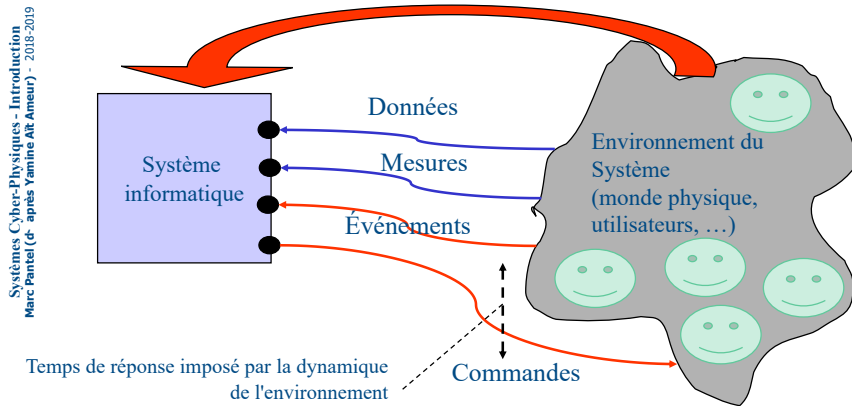


## Quelques exemples Fil rouge pour la matière : Le robot NXT Lego



## Le contexte général

### Rythme de l'interaction (systèmes temps réels)



## Systèmes temps réel

### ◆ Définition générale (CNRS 1988) :

◆ Peut être qualifiée de "**temps-réel**" (ou "**temps contraint**", ou encore "**réactif**")

- ❖ toute application mettant en œuvre un système informatique
- ❖ dont le fonctionnement est assujéti à l'évolution dynamique de l'environnement qui lui est connecté
- ❖ dont il doit contrôler le comportement.

## Les systèmes embarqués

### ◆ Les systèmes embarqués sont des systèmes

- ◆ temps réel
- ◆ critiques

### ◆ soumis aux problèmes classiques :

- ◆ des systèmes temps réel !
- ◆ De leur contexte d'utilisation

## Systèmes cyber-Physiques

### ◆ Définition

❖ Un **système cyber-physique (CPS)** est un système où des éléments informatiques collaborent pour le contrôle et la commande d'entités physiques

- Contrôle/commande (partie Cyber)
- Capteurs (lien Physique vers Cyber)
- Actionneurs (lien Cyber vers Physique)

### ◆ Définition de Edward Alan Lee (Univ. Berkeley)

Cyber-Physical Systems (CPS) are integrations of computation and physical processes.

Embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes affect computations and vice versa.

## Systèmes cyber-Physiques Les enjeux

### ◆ Spécification, Conception et Maintenance

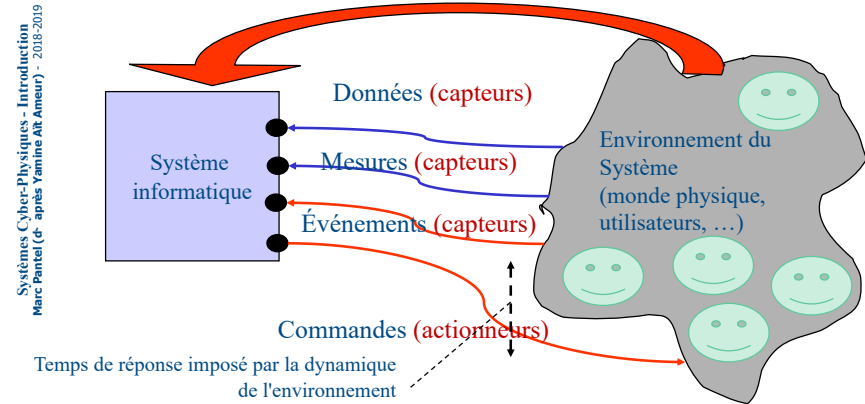
#### ◆ Composition

❖ Systèmes embarqués = Systèmes Cyber-physiques de 1<sup>ère</sup> Génération

- ◆ Programmation des CPS
- ◆ Sécurité & Sûreté
- ◆ Certification
- ◆ Sûreté de fonctionnement
- ◆ Energie & Environnement
  - ❖ Démantèlement
- ◆ Coûts de production

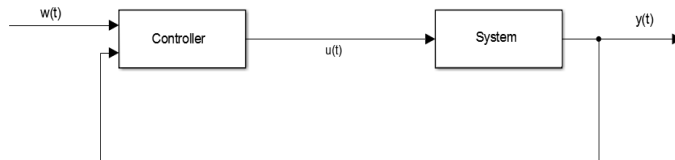
## Systèmes cyber-Physiques Spécification des besoins

### Rythme de l'interaction (systèmes temps réels)



## Systèmes cyber-Physiques Modélisation mathématique

### ◆ Modélisation mathématique



#### ◆ Attention : Nomenclature différente

- ◆ **Système : Environnement physique (avec capteurs/actionneurs)**
- ◆ U : Entrées du système à contrôler (commandes par actionneurs)
- ◆ Y : Sorties du système à contrôler (capteurs)
- ◆ X : Etat du système complet (externe et interne)
- ◆ W : Consigne (utilisateur)
- ◆ Equations qui régissent le système

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t), \mathbf{u}(t)) \end{aligned}$$

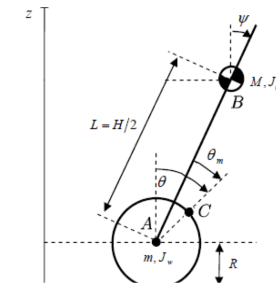
## Systèmes cyber-Physiques Modélisation Mathématique

### ◆ Cas du pendule inversé.

#### ◆ Modélisation mathématique

- ❖  $\theta$  angle de la roue
- ❖  $\theta_m$  angle moteur
- ❖  $\psi$  angle du pendule

- ◆  $\mathbf{x} = (\theta, \psi, \theta', \psi')^T$
- ◆  $\mathbf{y} = (\theta_m, \psi)^T$
- ◆  $\mathbf{u} = (v_l, v_r)^T$
- ◆  $\mathbf{w} = \theta_{ref}$



» Comment déterminer les pas de temps ?  
=> Discretisation

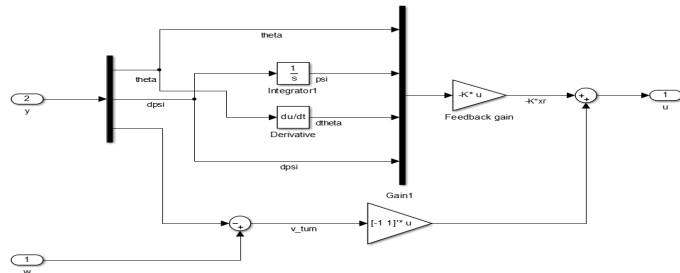
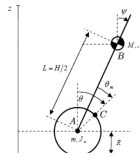


## Systèmes cyber-Physiques Conception, Validation et Implémentation

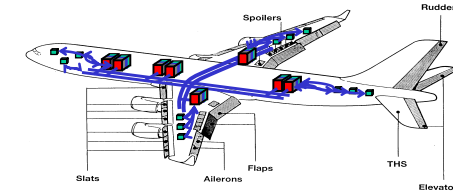
### ◆ Cas du pendule inversé.

#### ◆ Simulation du modèle mathématique

- ❖ Diagramme Simulink
- ❖ Génération de code du contrôleur



## ◆ Un exemple L' avionique Civile



## Pourquoi l'avionique ?

### ◆ Pourquoi le choix de l'avionique ?

#### ◆ Illustration parfaite pour les systèmes cyber-physiques

- ❖ Matériel
- ❖ Logiciel
- ❖ Réseaux
- ❖ Capteurs / Actionneurs

#### ◆ Avancées considérables

- ❖ Contrôle & commande
- ❖ Sécurité & sûreté
- ❖ Plusieurs innovations ces dernières années

## Pourquoi l'avionique ?

### ◆ Exemple : commande de vol d'un avion de combat

#### ◆ mission : stabiliser un avion *naturellement instable*

- lois de commande (calcul des ordres de gouvernes)
- asservissement des gouvernes

#### ◆ entrées = état et assiette de l'avion + ordres pilote

#### ◆ sorties = ordres de gouvernes + alarmes

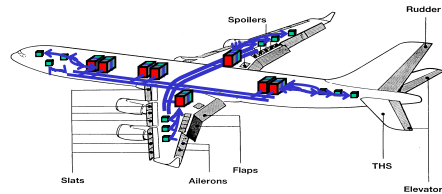
#### ◆ assujetti à la dynamique de l'avion :

- durée des boucles de commande = 10ms
- durée des boucles d'asservissement = 1ms

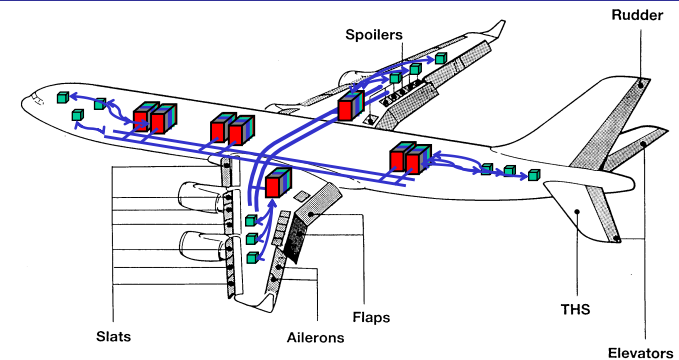
#### ◆ indispensable (et donc critique) : l'avion est incontrôlable sans ce système

- haut degré de **tolérance aux pannes**
- système **réparti** (plusieurs calculateurs reliés par un réseau constitué de bus de terrain)  
⇒ **garantie d'absence de bug !**

### ◆ L'avionique Civile



◆ Système avionique = ensemble des moyens informatiques embarqués à bord d'un avion, lanceur, satellite...



2018-2019

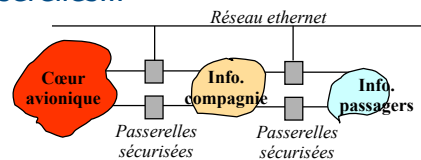
21

2018-2019

22

## Avionique civile

- ◆ logiciels d'application (pilote automatique...)
- ◆ logiciels exécutifs...
- ◆ calculateurs, bus, passerelles...



- ◆ cœur avionique =
  - gestion du vol (navigation, pilotage)
  - gestion mission (militaire)
- ◆ informatique compagnie
- ◆ informatique passagers

- ◆ Dans le cœur avionique, on trouve des systèmes de...

- ◆ **Guidage**
- ◆ **Pilotage**
- ◆ **Asservissement** des organes de pilotage (gouvernes et moteurs)
- ◆ **Protection** domaine de vol

- ◆ **Interface** avion / équipage

- ◆ Divers : gestion carburant, gestion alimentation électrique, anticollision...

2018-2019

23

2018-2019

24



### ◆ Commandes de vol

- ❖ Systèmes liant le pilote et les gouvernes aérodynamiques pour modifier la trajectoire de l'avion

### ◆ Évolution

- ❖ Premiers avions + avions légers
  - liaison entre le manche, le palonnier et les gouvernes de direction et de profondeur réalisée à l'aide de câbles
  - le pilote exerce directement sa force sur les gouvernes.
- ❖ Avions devenus plus lourds
  - Ajout des systèmes hydrauliques d'assistance permettant ainsi de démultiplier la force du pilote.

### ◆ Apparition de calculateurs numériques rapides

- ❖ => Modification de la tâche du pilote.
  - Il ne détermine plus le mouvement des gouvernes pour en contrôler ensuite les effets grâce à ses instruments de bord
  - Il détermine le mouvement de l'avion
    - » Le calculateur commande les mouvements de gouvernes nécessaires en fonction de l'altitude, de la vitesse etc.
    - » Pas d'effort physique
    - » Certains constructeurs ont pu remplacer le système traditionnel manche et palonnier par un simple mini-manche.
    - » Le calculateur peut interdire des mouvements de gouverne qui pourraient entraîner des risques pour la sécurité du vol.

### ◆ Commande de vol informatisée =

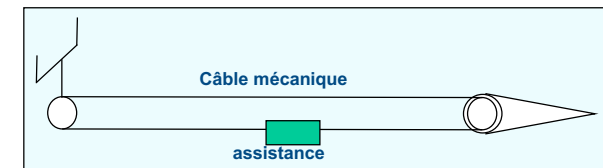
- ❖ des calculateurs redondants
  - mise en œuvre des lois de pilotage de l'avion,
- ❖ des règles de sécurité
  - limitant éventuellement les évolutions de l'avion à son enveloppe de vol, c'est-à-dire aux régimes et attitudes sûrs.

### ◆ Philosophie différentes Airbus et Boeing

- ❖ Boeing permet de transgresser ces limites par pression au-delà d'une butée,
- ❖ Airbus empêche toute évolution au-delà des limites de sécurité.
- ❖ Chaque approche a des arguments en sa faveur et sa défaveur.

Exemple.

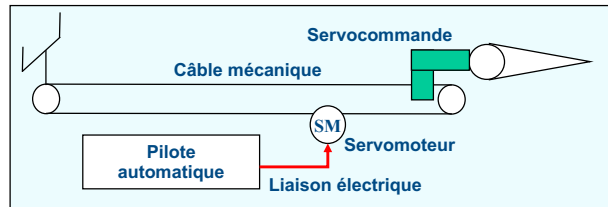
Évolution du système de contrôle du vol à court terme (FGS + FCS)...



Commande classique « à câble »

### Exemple.

Évolution du système de contrôle du vol à court terme (FGS + FCS)...



Commande classique « servomoteur » (A300 / A310)

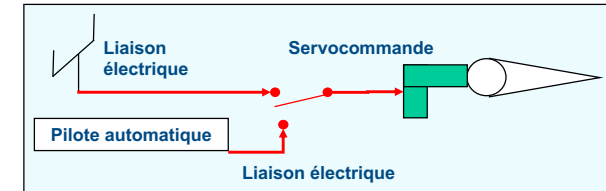
Un **servomoteur** (ou servo-moteur) est un moteur électrique conçu pour générer le mouvement précis d'un élément mécanique selon une commande externe.

Une **servocommande** est un système permettant de contrôler un mécanisme. Il permet un contrôle très précis de la position.

Les servocommandes utilisent le principe de retour de force négatif (*negative feedback*) : la valeur électrique reçue est comparée à la position du système contrôlé, en utilisant un potentiomètre intégré par exemple, et la différence de valeur est amplifiée puis utilisée pour actionner un moteur qui va corriger l'erreur (la différence).

### Exemple.

Évolution du système de contrôle du vol à court terme (FGS + FCS)...

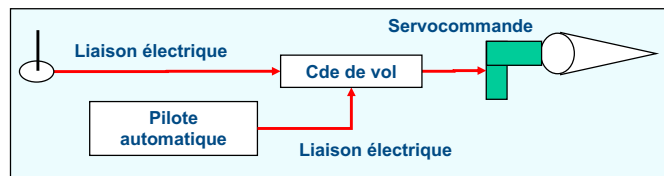


Commande électriques « lois directes »

Remarque : le Concorde possédait déjà (en 1969) des commandes de vols électriques

### Exemple.

Évolution du système de contrôle du vol à court terme (FGS + FCS)...



Commande électriques « lois normales » (A320 / A330 / A340 / A380 / A350 / A400M)

Les **commandes de vol électriques** sont une évolution des commandes de vol traditionnelles.

En aviation, les commandes de vol sont le système qui fait le lien entre le pilote et les gouvernes aérodynamiques qui permettent de modifier la trajectoire de l'avion.

Ce système est donc composé d'organes de pilotage (manche par exemple), d'actionneurs pour bouger les gouvernes, et d'un système de transmission plus ou moins sophistiqué entre les organes de pilotage et les actionneurs

### ◆ Description des étapes de développement

#### ◆ Définition d'un cycle de développement à plusieurs étapes

- ❖ Spécification
- ❖ Conception générale
- ❖ Conception détaillée

### ◆ Cas du système de contrôle des gouvernes de l'A320.



### ◆ Spécification

- ❖ => l'ensemble des exigences que le système doit satisfaire

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

- ❖ Le système doit traduire en ordres de déflexion des gouvernes les ordres de pilotage venant du pilote ou de pilote automatique (facteur de charge)
- ❖ Le système doit maintenir l'avion dans son domaine de vol quelles que soient les commandes du pilote ou du pilote automatique
- ❖ ...
- ❖ Le système doit être déterministe (fonctionnellement et temporellement)
- ❖ ...

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

- ❖ Compte tenu de la dynamique de l'avion, les ailerons et la gouverne de direction doivent être asservis à une période minimale **de 10ms**
- ❖ Compte tenu de la dynamique de l'avion, la gouverne de profondeur doit être asservie à une période minimale **de 30ms**
- ❖ ...
- ❖ La perte de contrôle à la fois des ailerons et des spoilers est catastrophique (taux de panne **de  $10^{-9}$** )
- ❖ La perte de contrôle de la profondeur est catastrophique (taux de panne **de  $10^{-9}$** )
- ❖ ...
- ❖ La perte de contrôle des spoilers (becquet), des ailerons, de la direction et de la profondeur ne doit pas être causée par une **panne unique**
- ❖ ...

### ◆ Conception générale

#### ◆ l'architecture du système :

- système réalisé en informatique

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

#### ❖ 9 calculateurs

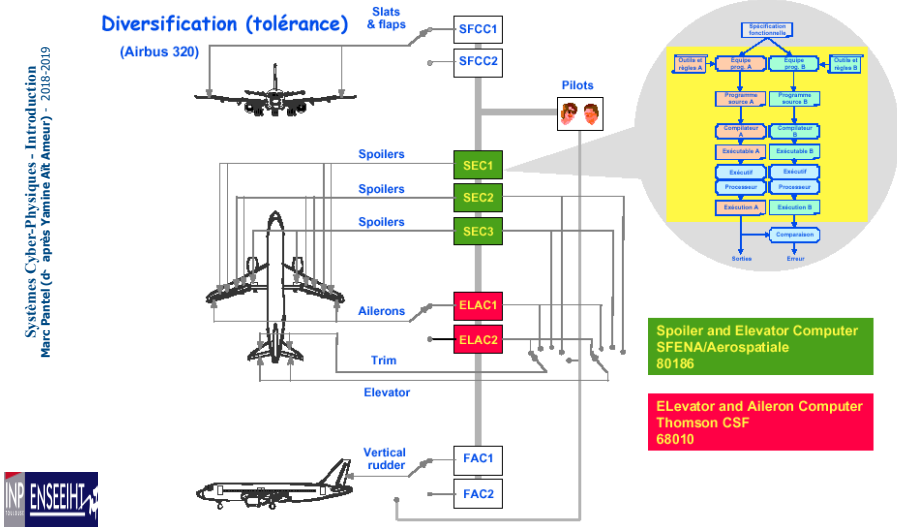
- » 2 calculateurs redondants pour les becs et les volets (SFCC1 et 2)
- » 2 calculateurs redondants pour la direction (FAC1 et 2)
- » 3 calculateurs redondants pour les spoilers, la profondeur et le trim (SEC1, 2 et 3)
- » 2 calculateurs redondants pour les ailerons, la profondeur et le trim (allure) (ELAC1 et 2), suppléés en cas de panne par les calculateurs des spoilers (SEC1, 2 et 3)

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

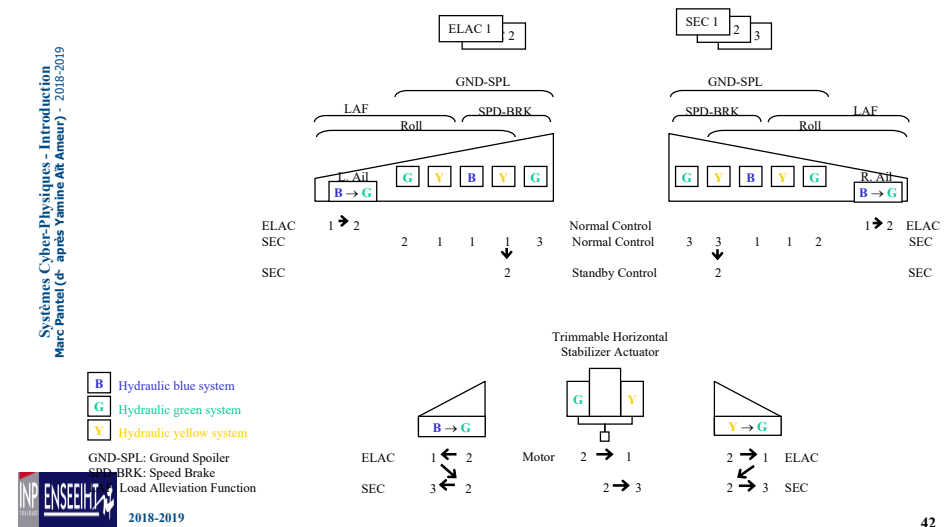
#### ❖ 9 calculateurs (suite)

- » Chaque calculateur doit être "fail-silent"
- » Chaque calculateur doit être caractérisé par un taux de panne inférieur ou égal à  $10^{-3}$  panne par heure de fonctionnement
- » ...

## ◆ Exemple : système de contrôle des gouvernes de l' A320...



## ◆ Exemple : système de contrôle des gouvernes de l' A320 ...

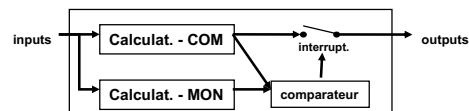


## ◆ Exemple : le système de contrôle des gouvernes de l' A320...

### ◆ Conception générale (suite)

❖ Choix de conception : chaque calculateur est de type COM - MON

- deux parties COM et MON fonctionnellement identiques mais réalisées différemment
- comparaisons des sorties des deux parties
  - » si accord, alors émission des sorties vers l'extérieur
  - » si désaccord, alors le calculateur est isolé de son environnement (plus aucune sortie)



## ◆ Exemple : le système de contrôle des gouvernes de l' A320...

### ◆ Conception générale (suite)

❖ Choix de conception :

■ Dans chaque calculateur : identification des tâches :

- » de calcul des ordres de déflexion de gouverne (lois de pilotage)
- » d'asservissement des gouvernes (lois d'asservissements)
- » (une tâche) de surveillance et consolidation des entrées
- » de logique d'engagement

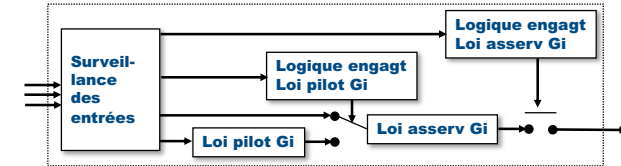
### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

#### ◆ Conception générale (suite)

##### ❖ Choix de conception :

##### ■ Calculateurs SEC1, 2 et 3

- » une **tâche de calcul des ordres de déflexion du spoiler X** de **période = 40 ms**
- » une **tâche de calcul des ordres de déflexion de la profondeur** de **période = 40 ms**
- » ...
- » une **tâche d'asservissement du spoiler X** de **période = 10 ms**
- » une **tâche d'asservissement de la profondeur** de **période = 20 ms**
- » ...
- » une **tâche de surveillance et consolidation des entrées** de **période = 10 ms**
- » une **tâche de logique d'engagement de la tâche de calcul des ordres de déflexion du spoiler X** de **période = 120 ms**
- » ...



### ◆ Conception détaillée

#### ◆ Définitions plus précises et choix d'algorithmes

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

##### ❖ Choix de conception :

- chaque tâche sera codée en **SCADE/LUSTRE (un langage synchrone)**
- choix **d'un ordonnancement statique** (cyclique)
- => découpage du temps en **tranche de 10 ms** et allocation (statiques) des tâches dans ces tranches
  - » les tâches de périodes 10 ms sont exécutées dans toutes les tranches
  - » les tâches de périodes 20 ms sont exécutées dans une tranche sur deux...

### ◆ Conception détaillée

#### ◆ Définitions plus précises et choix d'algorithmes

### ◆ Exemple : le système de contrôle des gouvernes de l'A320...

##### ❖ Choix de conception (suite) :

- => seule **IT** prise en compte = **top d'horloge** (toutes les 10 ms)
- => l'ordonnancement statique sera codé en **C**
- => pas d'OS temps réel

##### ❖ etc.

## Évolution : croissance continue

	1980 A310	1985 A320	1990 A340
Taille totale du logiciel embarqué (en Mo)	4	10	20
Nombre total d'équipements (calculateurs)	77	102	115
Nombre de bus numériques	136	253	368
Volume total (en litres) de l'électronique embarquée	745	760	830
Volume (en litres) de l'électronique du pilote automatique	134	63	31

❖ L'avionique (ensemble des équipements numériques et des logiciels) représente entre **33%** du coût d'un avion

❖ la phase de validation de l'avionique représente environ **50%** du temps de développement total

## ◆ A partir de A380 : avionique modulaire intégrée (IMA)

### ◆ Pourquoi de nouvelles architectures ?

- augmentation du nombre de fonctions
- intégration croissante des systèmes
- augmentation des échanges entre les fonctions
- augmentation des besoins en ressources de calcul et de communication

### ◆ Nécessité de partage des ressources

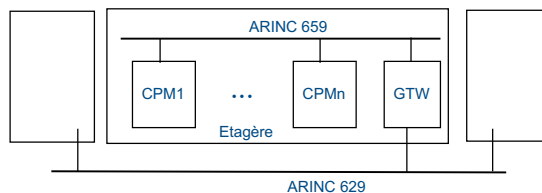
- bus de **multiplexés**
- ressources de traitement gérées par un **exécutif temps réel**
- ressources de calcul et de communication **banalisées**

### ◆ 3 types d'IMA

- Boeing (B777)
- Airbus (A380)
- Dassault Aviation (Rafale)

## ◆ IMA Boeing (B777)

- à base de calculateurs partagés (CPM)
- à base de bus partagés CSMA-CA (ARINC 629 et ARINC 659)
- ordonnancement statique des fonctions sur les CPM et des messages sur les bus



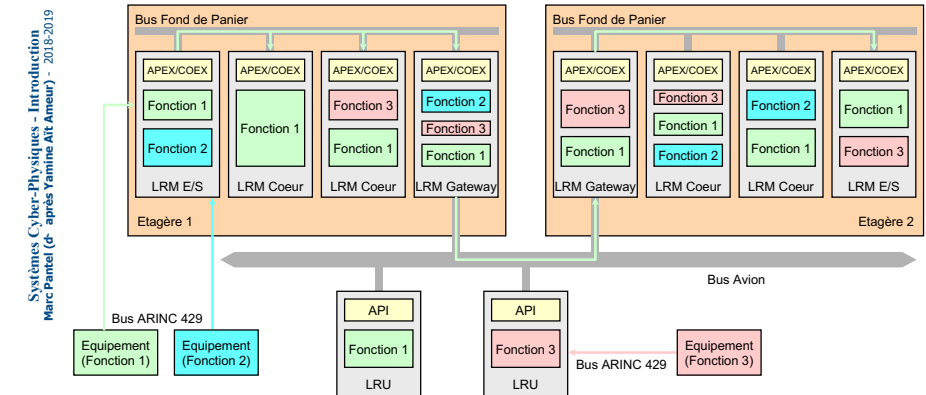
### ◆ Intérêt

- déterminisme

### ◆ Principaux problèmes

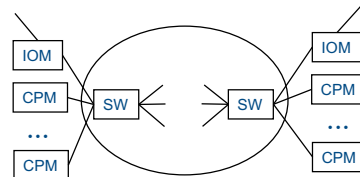
- définition et maintenance des trames bus
- manque de souplesse, et évolution difficile

## ◆ Architecture IMA : Type Boeing (B777)



### ◆ IMA Airbus (A380) :

- à base de calculateurs partagés (CPM)
- interconnectés par un réseau en étoile composé de commutateurs (SW) et de bus Ethernet
- ordonnancement statique des fonctions sur les CPM et des messages sur les bus



### ◆ Intérêt

- grande souplesse => facilité d'évolution
- composants du commerce => faible coût et maintenance plus aisée

### ◆ Principaux problèmes

- évaluation de performances et dimensionnement du réseau (risque d'engorgement du réseau)
- problème de garantie de déterminisme

### ◆ Les systèmes avioniques sont **critiques** (leur défaillance met en danger l'équipage ou les passagers)

- ❖ => besoin de **déterminisme**
- ❖ => besoin de **sûreté** (pas de défaillance)
- ❖ => besoin de **sécurité** (robustesse aux malveillances)
- ❖ => besoin de **disponibilité** (système essentiels)

### ◆ => **Conséquences architecturales**

- redondance matérielle avec dissimilarité (matérielle et logicielle)
- ségrégation géographique

### ◆ => **Conséquences fonctionnelles**

- déterminisme (SCADE/Lustre, allocations statiques, séquençements statiques...)
- tolérance à l'asynchronisme de l'architecture

### ◆ => **Besoin fort en**

- ❖ garantie de correction fonctionnelle (absence de bug)
- ❖ analyse de sûreté de fonctionnement
- ❖ analyse de performance temps réel (garantie de non perte de message, de latence max...)
- ❖ => besoin en garantie de qualité de service
- ❖ => nécessité de démonstrations formelles pour le dossier de certification

### ◆ Un avion est un système complexe qui fait intervenir plusieurs partenaires

» besoin de conception, développement, vérification, réalisation... **séparés** !

### ◆ Les campagnes d'essai en vol sont longues et chères, et demandent de nombreuses modifications sur le logiciel embarqué

- ❖ besoin de réactivité lors de modification du code embarqué
  - » (délai exigé entre une demande de modification et l'instant où l'avion est prêt = quelques heures)
- ❖ besoin de techniques de modifications incrémentales
- ❖ besoin de techniques de vérification formelle et exhaustives
- ❖ besoin de techniques de génération de jeux de test...



## Avionique civile : développement

### ◆ Un avion civil doit être certifié...

- ◆ Obligation de constituer un dossier de certification qui doit être accepté par les autorités européenne... et américaine !
  - dossier volumineux et long (donc coûteux) à établir
  - chaque modification doit donner lieu à une re-certification
  - la certification est un frein aux évolutions...
  - ... à moins de disposer de **techniques formelles** (mathématiques) permettant de démontrer **rigoureusement** et **rapidement** la conformité d'une évolution par rapport aux exigences et sûreté

## Avionique civile : conclusion

### ◆ => Les techniques nécessaires au développement de systèmes avioniques

- ❖ ingénierie système et ingénierie logiciel
- ❖ automatique discrète et continue
- ❖ spécification et vérification formelles
- ❖ sûreté de fonctionnement
- ❖ analyse de performance temps réel
- ❖ réseaux
- ❖ ...

## Avionique civile : conclusion

### ◆ Remarque : coût de vente d'un avion civil :

- 1/3 cellule
- 1/3 moteurs
- **1/3 avionique**

#### ❖ Mais

- part de plus en plus importante (prépondérante ?) de l'avionique dans l'amélioration des performances de l'avion

#### ❖ Exemples :

- » structure légère rendue possible les commandes de vol électriques (FCS)
- » avion instable rendu possible par la qualité des commandes de vol électriques (FCS)
- » minimisation de la consommation de carburant par une meilleure gestion du vol (FMS) => vers le « vol libre » ?
- » ...

## Avionique civile : conclusion

◆ avionique = enjeu stratégique pour les avionneurs !

### ◆ Mais aussi pour

- ❖ L'espace
- ❖ L'automobile, le ferroviaire
- ❖ L'énergie et l'environnement
- ❖ La médecine
- ❖ Les télécommunications,
- ❖ ...

## ◆ Conclusion

## Liens avec les autres cours du cursus ingénieur

- ◆ Mathématiques et modélisation
  - ◆ Continu/discret
    - ❖ Discrétisation
- ◆ Conception de programmes corrects et sûrs de fonctionnement
  - ❖ Absence de bug
  - ❖ Validation, vérification, simulation,
- ◆ Respect des délais
  - ❖ Systèmes temps réels
  - ❖ Délais de bout en bout dans les réseaux
- ◆ Prise en compte du volume de données
  - ❖ Exemple de la vidéo dans des drones

## A venir

- ◆ Mathématiques et modélisation
- ◆ Simulation
- ◆ Programmation
- ◆ Avec des
  - ◆ TD
  - ◆ TP
  - ◆ BE