

# Mécatronique des véhicules automobiles - La norme AUTOSAR

**L. NEHAOUA**

lamri.nehaoua at ufrst.univ-evry.fr

**M2 GEII, Cours 2, 10/10/2013**

## Contexte et problématique

- L'automobile : 3e secteur d'activité multi-disciplinaire.
- plus de 20% de la valeur d'une voiture est composée de systèmes embarqués.
- motorisation, confort, multimédia, interactions, assistance, autonomie et sécurité.
- l'augmentation de la part des SE induit un accroissement:
  - ▶ de la complexité de conception,
  - ▶ des coûts de conception,
  - ▶ compétitivité: accélérer ses processus d'innovation.
- pôles de compétitivité:
  1. Mov'eo.
  2. ID4Car.
- quelques équipementiers:
  1. France: Faurecia, Valeo.
  2. Allemagne: Bosch, Siemens VDO Automotive, Continental AG, ZF Friedrichshafen.
  3. USA: Delphi, TRW Automotive.
  4. Japon: JTEKT, Denso, Aisin Seiki, Yazaki, Toyota Boshoku, Sumitomo Electric.

## Contexte et problématique

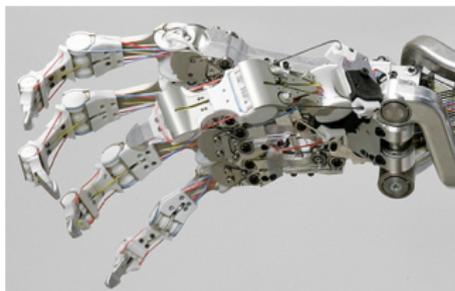
- objectif:
  - ▶ contrôle optimal du véhicule: actionneurs, capteurs et calculateurs.
  - ▶ améliorer les performances de la conduite et de la manoeuvrabilité.
  - ▶ prédire les situations de risque et les perturbations externes.
  - ▶ diagnostic, tolérance aux défauts et robustesse des systèmes du contrôle (en 2003, en Allemagne, la défaillance des systèmes électriques est en cause de 49,2% des pannes).
- défis: garantir une parfaite intégration de l'ensemble des systèmes mécatroniques.
  - ▶ dynamiques variables: châssis (lente et quasi-linéaire), contact pneu-sol (rapide et non-linéaire) → contrôle multi-rates, multi-physiques → contrôle multi-couches.
  - ▶ contrôle global distribué ou local interconnecté?
  - ▶ limitation: capteurs, actionneurs, systèmes embarqués → optimisation multi-critères.

## Contexte et problématique

- exigences législatives: lois gouvernementales (environnement)
  1. réchauffement : niveau d'émission de gaz,
  2. pollution : émissions de particules (Diesel),
  3. consommations : besoins stratégiques (dépendance énergétique),
- exigences du client: nouveaux services.
  - ▶ confort: mieux voyager,
  - ▶ sécurité: mieux protégé,
  - ▶ consommations: moins dépenser,
  - ▶ coût: moins investir,
- exigences du constructeur : développement facile.
  1. prototypage: gain de temps, HIL,
  2. coût: réduction des coûts,
  3. gestion: time to market,
- véhicule de rêve: sécurité en situations extrêmes ou éviter les embouteillages.

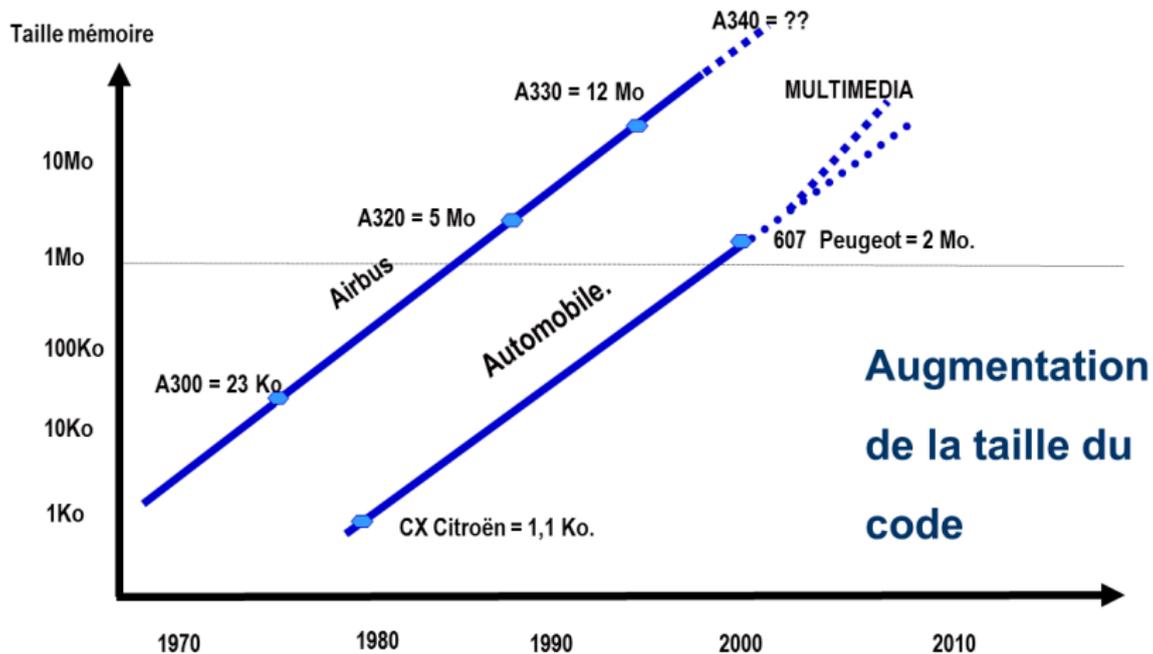
## mécatronique automobile

- Mécatronique : l'intégration synergique de la mécanique, l'électronique et l'informatique pour produire des dispositifs intelligents.



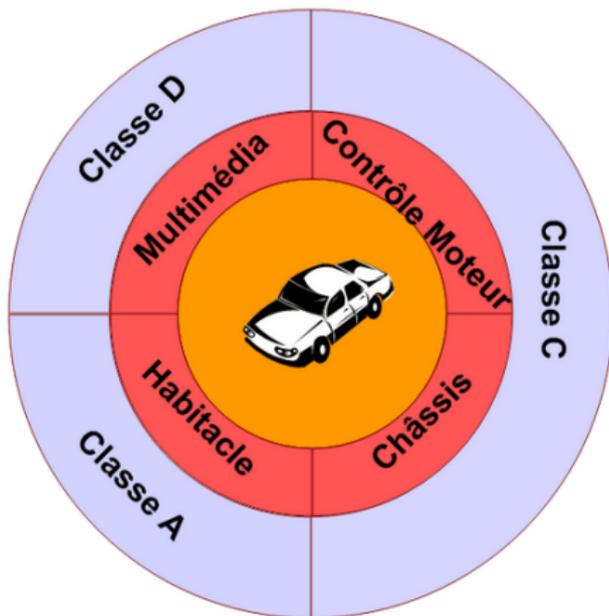
- Intrusion des systèmes mécatroniques:
  - ▶ Volkswagen Passat, 10000 composants électriques, 61  $\mu$ -processeurs et 3 réseaux CAN.
  - ▶ Mercedes S-Class: au moins 70 ECU.
  - ▶ BMW S7: 80  $\mu$ -processeurs.
  - ▶ Caterpillar 797: système embarqué distribué de 195 capteurs/actionneurs et transmission sans fil.
- flux de données:
  - ▶ Citroën CX (1980): 1.1KByte,
  - ▶ Peugeot 607 (2000): 2MByte,
  - ▶ 2004: 10MByte.

## mécatronique automobile



## mécatronique automobile

- historiquement : 5 domaines sont identifiés.
  1. chaîne de traction : propulsion/traction,
  2. châssis : direction et freinage,
  3. habitacle : airbags, éclairage, vitre, air-conditionnée, équipement siège,
  4. IHM : échanges conducteur/véhicule (affichage et sélection),
  5. télématique : échanges véhicule/environnement (navigation, radio, paiement),



## mécatronique automobile

1. chaîne de traction: ACC, injection, ...
  - ▶ lois de contrôle complexes,
  - ▶ importante puissance de calcul (co-processeurs,  $\mu$ P 16/32 bits, DSP).
  - ▶ contraintes de temps strictes
  - ▶ multi-rate: temps moteur: 0.1ms, temps déplacement: 1-5ms.
2. châssis: ABS, ESP, 4WD, ...
  - ▶ de même,
  - ▶ multi-rate: 0.1ms-100ms.
3. habitacle :
4. multimédia/télématique :
  - ▶ bande passante.
  - ▶ partage des ressources.
  - ▶ QoS et sécurité.
  - ▶ ergonomie et acceptabilité.

## mécatronique automobile

- Pourquoi un bus de communication?
  1. implémentation (stand-alone ECU) :  $\mu$ C + capteurs + actionneurs,
  2. inadaptée pour une fonctionnalité distribuée (une fonction utilisant plusieurs ECU),
  3. exemple: plusieurs ECU utilisent la mesure de la vitesse.
  4. 2500 variables de mesures partagées par 70 ECU.
- Les bus sont classés suivant plusieurs critères:
  1. liaison.
  2. technologie.
  3. domaine d'utilisation.

## mécatronique automobile

### 1. liaison:

- 1.1 point-par-point :  $n^2$  canaux pour  $n$  ECU (poids, coût, fiabilité).
- 1.2 multiplexé : communications multiplexées sur un support partagé.

### 2. technologie:

- 2.1 bus prioritaires: CAN (Controller area network), VAN (Vehicle area network), J1850 (norme SAE).
- 2.2 bus TT (TTEthernet): FlexRay, TTCAN.
- 2.3 bus Low-Cost: LIN (Local Interconnect Network), TTP (Time-Triggered Protocol).
- 2.4 bus Multimédia: MOST (Media Oriented Systems Transport), IDB-1394 (version automobile de IEEE 1394 - Firewire)

### 3. domaine d'utilisation:

- 3.1 classe A: pour l'habitacle. Débit < 10kbps. Bus LIN et TTP/A,
- 3.2 classe B: pour l'habitacle. Débit de 10-500 kbps. Bus CAN-B,
- 3.3 classe C: contrôle moteur et du châssis. Débit < 1 Mbps. Bus CAN-C, TTP/C, FlexRay. Grande sécurité, fiabilité et fonctions de diagnostic.
- 3.4 classe D: multimédia et télématique. Débit de 25-150Mbps. Bus MOST.

## mécatronique automobile

- MOST:

- ▶ créé pour des applications multimédia dans l'automobile. Depuis 1997.
- ▶ MOST25 (25Mbps, fibre optique, long. trame 512 bits), MOST50 (50Mbps, 1024 bits), MOST150 (150Mbps, 3072 bits)
- ▶ 1 maître / plusieurs esclave avec une topologie token-ring.

- FlexRay:

- ▶ BMW, Bosch, Daimler, GM, NXP Semiconductors, Freescale et VW.
- ▶ 1ere publication 2004 et version 2.1 en 2005.
- ▶ débit: 10 Mbps sur 2 canaux (A et B) soit 20 Mb/s.
- ▶ longueur trames 254 octets.
- ▶ mécanismes de tolérance aux fautes.
- ▶ un bus déterministe. Technique TDMA (Time-Division Multiple Access).

- TTP:

- ▶ développé par TTTech.
- ▶ bus déterministe. Technique TDMA (Time-Division Multiple Access).

## Norme Autosar

- complexité des architectures électroniques embarquées:
  - ▶ diversité des réseaux,
  - ▶ diversité des calculateurs,
  - ▶ gestion de l'obsolescence, de la durée de vie et de la migration.
- besoin d'harmonisation dans la conception des applications embarquées.
- l'objectif est de s'affranchir du matériel lors de la conception.
- solution: standard
  - ▶ consortium: Audi, BMW, Daimler-Chrysler, Porsche, Volkswagen, Siemens et Bosch.
  - ▶ concevoir des logiciels de base communs aux ECU de tous les fabricants,
  - ▶ abstraction du matériel tout en conservant des applications propriétaires.
- autres problèmes:
  - ▶ standardiser les logiciels → abstraction du l'OS.
- Solution finale: Autosar (AUTomotive Open System ARchitecture).
  - ▶ se concentrer sur les applications et non plus sur la manière de les intégrer à un véhicule.
- méthode proposée: réorganiser les couches basses des logiciels ECU sous forme de modules
  - ▶ le développement d'une couche d'abstraction du matériel (modèle d'ECU, type de réseau, ...)
  - ▶ la définition de l'architecture logicielle de base.

## Norme Autosar

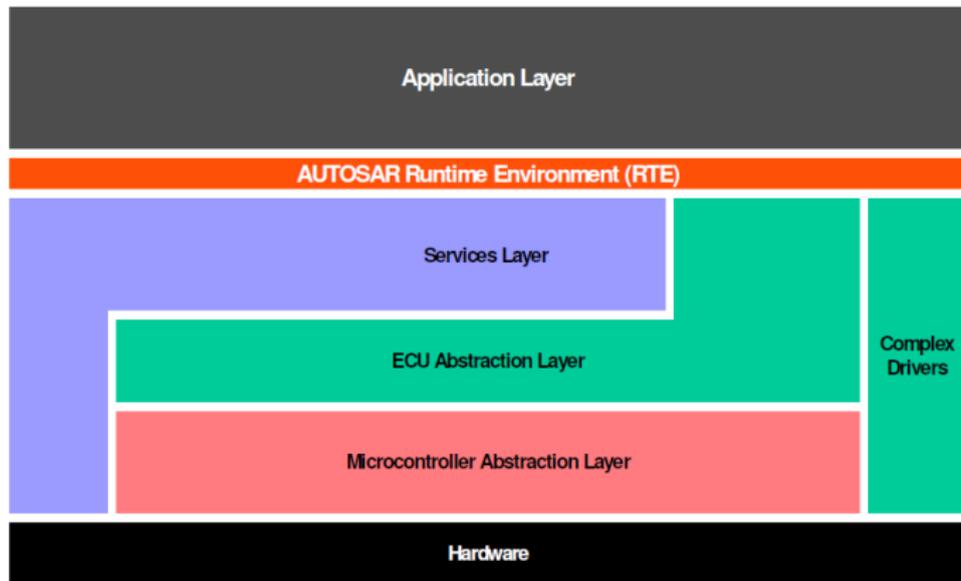


Figure : architecture globale en couches de la norme AUTOSAR

## Norme Autosar

### 1. RTE (Runtime Environment)

- ▶ permet la programmation et l'exécution d'une fonction indépendamment de l'architecture du calculateur (ECU)
- ▶ définir un bus virtuel (Middleware) qui assure des services de communication entre les composants logiciels (SWC) et la couche application.
- ▶ ces communications peuvent être inter-ECU ou intra-ECU.

### 2. BSW (Basic Software)

#### 2.1 Services Layer:

- ▶ gère les services à intégrer au calculateur (protocoles, gestion de la mémoire et diagnostic de l'état de l'ECU).

#### 2.2 Communication Layer:

- ▶ framework d'abstraction de communication (CAN, LIN, FlexRay, ...), gestion d'E/S et gestion réseau.

#### 2.3 ECU Abstraction Layer:

- ▶ couche d'abstraction du calculateur global (processeur+chipsets):.
- ▶ permet de découpler la couche haute de l'application et la couche basse de l'ECU (spécifications électriques).

## Norme Autosar

### 2.4 Complex Device Driver:

- ▶ assurer l'accès direct au  $\mu$ C. Cet accès est nécessaire pour les algorithmes de contrôle.

### 2.5 Microcontroller Abstraction Layer:

- ▶ permet d'éviter l'accès direct au registre du processeur.
- ▶ gestion bas-niveau (driver) du  $\mu$ C et de ses périphériques (DIO, ADC, PWM, EEPROM, WDT, SPI, I2C, ...).

### 2.6 Operating System:

- ▶ spécifier les exigences du système d'exploitation.
- ▶ performance temps-réel.
- ▶ ordonnancement basé sur la notion de priorité.
- ▶ exception: quelques domaines (télématique) utilise leur propres OS. Dans ce cas l'interface avec les composants Autosar (SWC) doit être compatible avec Autosar (l'OS utilisé doit être converti en un Autosar OS).
- ▶ Autosar OS utilise le standard OSEK OS. ISO 17356-3: Véhicules routiers - Interface ouverte pour applications automobiles embarquées - Partie 3: Système d'exploitation OSEK/VDX (139 €)

## Norme Autosar

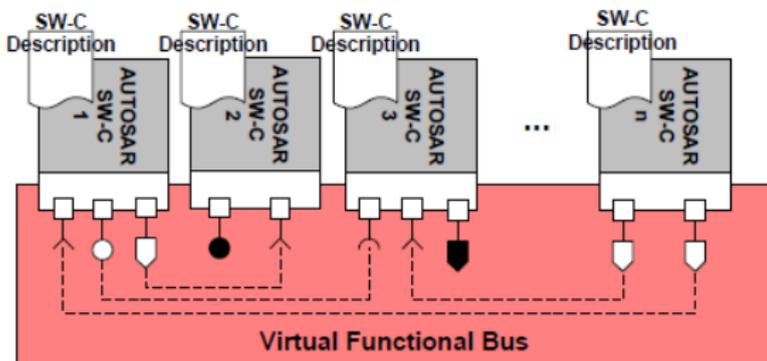
Le principe d'Autosar: changer la manière de programmer les calculateurs.  
L'implémentation de fonctions automobiles se fera en trois étapes:

### 1. SW-C (Software Components)

- ▶ les fonctions sont décrites de manière formelle comme étant des composants.
- ▶ déclarer les variables d'E/S indépendamment du matériel.

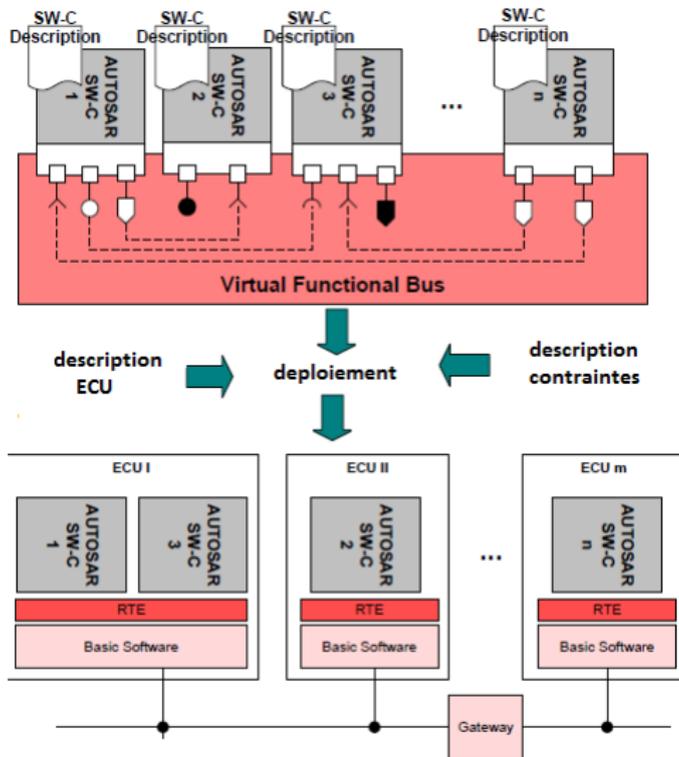
### 2. VFB (Virtual Functional Bus)

- ▶ générer des bus virtuels.
- ▶ ce bus relie l'ensemble des E/S. Toutes les fonctions doivent disposer de toutes les informations requises.



## Norme Autosar

- ajouter la description physique des calculateurs:
  - cela permet de générer une vraie architecture automobile.



# Norme Autosar

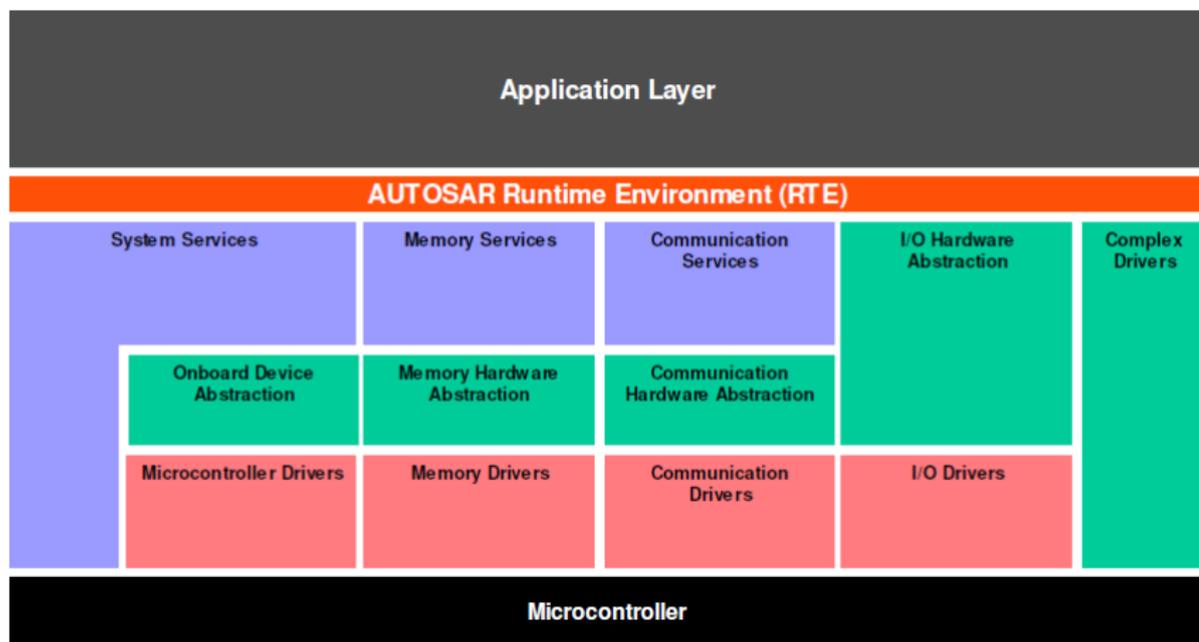


Figure : La couche Basic software contient plus de 80 modules.

## Norme Autosar

- MCAL est divisée en 4 parties:
  1. I/O drivers: ADC, PWM, DIO,
  2. Communication Drivers: pour l'ECU (ex. SPI, I2C) et pour le véhicule (ex. CAN),
  3. Memory Drivers: mémoire interne (ex. EEPROM) et externe (ex. Flash)
  4.  $\mu$ C Drivers: pour périphériques internes (ex. WDT, Clk) et fonction avec accès direct au  $\mu$ C (ex. test RAM)
- ECU-AL est divisée en 4 parties:
  1. I/O Hardware Abstraction: des modules pour représenter les signaux d'E/S tels qu'ils sont connectés à l'ECU (courant, voltage, fréquence). Cette couche ne fait l'abstraction des capteurs/actionneurs.
  2. Communication Hardware Abstraction: des modules pour garantir l'accès au bus indépendamment de son emplacement.
  3. Memory Hardware Abstraction: des modules pour garantir l'accès à la mémoire indépendamment de son emplacement.
  4. Onboard Device Abstraction: contient des drivers pour les composants intégrés à l'ECU (watchdog externe, timer externe, etc).

## Norme Autosar

- Service Layer est divisée en 3 parties:
  1. Communication Services: des modules représentant les bus réseaux du véhicule (CAN, LIN, FlexRay and MOST). Leur tâche est de fournir une interface uniforme aux bus réseaux pour l'échange des données, services, diagnostic.
  2. Memory Services: des modules pour gérer la mémoire non-volatile (NVRAM),
  3. System Services: des modules utilisés par les modules de l'ensemble des couches. Permet de fournir des services en relation avec l'OS.