



GRANDE ECOLE D'INGENIEURS GENERALISTES
Angers, Paris, Shanghai & réseau international
Siège : 4 rue Merlet de la Boulaye - BP 30926
49009 Angers cedex 01 - France
Tél.: +33 (0)2 41 86 67 67
Fax: +33 (0)2 41 87 99 27
information@eseo.fr - www.eseo.fr



geensys

La Fleuriaye
Rue Alessandro Volta
BP 50758
44481 CARQUEFOU cédex
FRANCE

Tél. : +33 (0)2 53 46 00 70
Fax : +33 (0)2 53 46 00 71
info@geensys.com

RAPPORT DE STAGE

TYPE DE STAGE : (*cochez la case correspondante*)

Rapport de parrainage ingénieurs

Stage Découverte (*précisez l'année P1, P2 ou I1*) :

Stage Scientifique & Technique

Stage de Fin d'Etudes

AUTEUR : Clément DAVID

DATES : Du 2 mars au 4 septembre 2009

NIVEAU DE CONFIDENTIALITE (*Concerne les stages en I2 et I3 uniquement*)

0 (aucune)

Niveau I

Niveau II

Niveau III



SUJET MISSIONS

Intégrer une pile TCP/IP open-source dans
l'architecture AUTOSAR

Entreprise : Geensys - Carquefou

Resp. de stage : Julien MONSIMIER, ingénieur développement en
systèmes embarqués

Remerciements

Je remercie toutes les personnes qui m'ont permis de mener à bien ce stage et plus particulièrement :

- Julien MONSIMIER, mon responsable de stage qui m'a accompagné et encadré tout au long de ce stage.
- Jean-Sébastien BERTHY, responsable d'AUTOSAR Lab.
- Aymeric VRIGNON, intégrateur sur le projet ARPOD pour ses conseils avisés.
- Luc FOREST avec qui j'ai eu le plaisir de travailler pour le projet ARPOD.

Ainsi que toute l'équipe d'AUTOSAR Lab et des autres services que j'ai côtoyé, pour l'accueil chaleureux et l'entente cordiale qu'il y a eu durant ces 6 mois.

Sommaire

Sommaire.....	3
Index des illustrations et tables.....	4
Fiche de synthèse.....	5
La société et son marché.	6
Historique.....	8
Ayrton Technology.....	8
TNI Software.....	9
Organisation actuelle de la société.....	11
Partenaires et Références.....	12
Geensys à Carquefou.....	13
Contexte du stage.....	15
Le consortium AUTOSAR.....	16
Objectif final du stage.....	18
Contraintes.....	19
Équipe en charge de l'intégration.....	20
Planning initial.....	22
Travail effectué.....	23
Critères de choix de la pile TCP/IP.....	24
Synthèse de la spécification.....	24
Une pile qui cible l'embarqué.....	25
Contraintes supplémentaires.....	26
Architecture choisie.....	27
Le module TCP/IP (TcpIp).....	28
Le module Socket Adaptor (SoAd).....	32
Le module Ethernet Driver (EthDrv).....	33
Critiques de cette architecture.....	33
L'application de démonstration.....	35
Contraintes posées.....	36
Retours d'expérience.....	37
Carte EVK1100 : Atmel AVR32 famille UC3A.....	37
Développement du module suivant la méthode MCSE-Geensys.....	40
Carte MPC5668 : Freescale PowerPC famille MPC55xx/MPC56xx.....	41
Conclusion.....	43
Glossaire.....	44
Bibliographie.....	45

Index des illustrations et tables

Index des illustrations

Présence de Geensys en France et dans le monde.....	6
Part de l'activité de Geensys par domaine.....	7
Logo d'Ayrton Technology.....	8
Logo de TNI Software.....	9
Organisation et responsabilités dans l'entreprise.....	11
Photo du site de Carquefou.....	13
Flux de compétences pour l'automobile.....	14
Logo du pôle ID4CAR.....	15
Logo du consortium.....	16
Membres du consortium AUTOSAR par niveaux de participation et compétences.....	16
Architecture simplifiée d'un ECU AUTOSAR.....	17
Extrait de la documentation ElektroBit/AUTOSAR supportant Ethernet.....	18
Organigramme de la solution Ethernet-AVR32.....	20
Organigramme de l'intégration pour ARPOD.....	21
Diagramme d'une pile TCP/IP générique.....	25
Architecture des couches basses AUTOSAR supportant TCP/IP.....	27
Logo du projet lwIP.....	29
Vue externe du module TcpIp.....	30
Décomposition en fichiers de la pile lwIP.....	31
Couverture du cycle en V lors de la planification d'un module.....	32
Les différentes intégration du driver Ethernet.....	33
Logo de la société Autocruise, projet ARPOD.....	35
Topologie réseau utilisée dans le projet ARPOD. Le réseau Ethernet est ici représenté en rouge, le réseau CAN en vert et l'alimentation en bleu.....	35
Graphe de dépendance de la pile lwIP.....	38
Graphe de dépendance pour la solution AUTOSAR - lwIP.....	39
Schéma d'intégration de la pile TCP/IP pour MPC5668.....	41
Diagramme de l'architecture à l'intégration du noeud 007.....	42

Index des tables

Fiche de synthèse de stage.....	5
Chiffres clés en 2008 pour Geensys.....	7
Références par 'Business Unit'.....	12

Fiche de synthèse

Contexte	<p>AUTOSAR (AUTomotive Open System ARchitecture) est le standard émergeant pour le développement de logiciels dans l'automobile. Un consortium de différents industriels a été formé et spécifie les différents modules logiciels existants dans les ECU¹.</p> <p>Le standard AUTOSAR 3.x possède une pile de communication contenant les modules de communication pour réseaux CAN, FlexRay et LIN. Dans la prochaine version majeure du standard (4.0), il est prévu d'intégrer la communication par Ethernet à travers l'ajout de fonctionnalités basées sur TCP/IP².</p>
Objectifs	<ul style="list-style-type: none"> • Intégrer une pile TCP/IP existante comme un module AUTOSAR. Ceci en tenant compte des spécificités du monde de l'embarqué. • Valider par l'implémentation les spécifications du consortium. • Définir une application de test démontrant les avantages de cette intégration aux constructeurs et équipementiers par rapport aux solutions employées actuellement.
Contraintes	<ul style="list-style-type: none"> • Cette implémentation devra respecter au maximum les spécifications ou un sous-ensemble des spécifications. • Les différents modules nécessaires à l'ajout de cette pile devront suivre les interfaces et les configurations définies dans les spécifications. • L'ensemble des modules devra être implémenté de façon portable et multi-architecture.
Rôle	<p>Je suis en charge de la conception, de l'implémentation et de la validation des différents modules nécessaires à l'intégration de la pile TCP/IP dans AUTOSAR.</p> <p>Les spécifications doivent aussi être complétées par des remarques lors de ces différentes phases. Une communication avec les personnes en charge de la spécification a aussi été nécessaire.</p>
Bilan	<p>Une première phase d'implémentation des modules a été réalisée dans le délai imparti sur cible Atmel AVR32. Cette phase consistait à valider rapidement les choix du consortium sur une cible générique.</p> <p>Une deuxième phase d'intégration a ensuite été menée sur cible Freescale MPC5668. Cette intégration est réalisée dans le cadre d'un projet client et nécessite une validation complète de la solution Geensys-AUTOSAR.</p>

Tableau 1: Fiche de synthèse de stage

1 Electronic Control Unit : Système ou Sous-Système électronique généralement basé sur un micro-contrôleur.

2 Une pile TCP/IP est un composant logiciel implémentant les couches de communications du modèles OSI.

La société et son marché

Geensys est un acteur spécialisé du monde de l'embarqué, centré sur le conseil et l'assistance à projet de haut niveau, les outils et les services méthodologiques à forte valeur ajoutée, les blocs d'IP, et des services d'ingénierie électronique-mécatronique.

Elle se distingue par une expertise méthodologique transverse et sectorielle forte ainsi que par des capacités d'exécution plurielles. Société privée, née de la fusion de deux des sociétés françaises du secteur de l'embarqué, Geensys possède une présence directe au plan international, au travers de ses bureaux de vente en France, en Allemagne, en Chine et au Japon et d'un réseau de partenaires et de distributeurs aux USA et en Europe.

La couverture mondiale du groupe Geensys est garantie par un important réseau de partenaires, comprenant des sociétés fournissant des solutions intégrés à leurs clients mais aussi des entreprises leaders dans leurs domaines (MathWorks, Artisan Software). Un important réseau de distributeurs permet aux produits Geensys d'être utilisés dans le monde entier.



Illustration 1: Présence de Geensys en France et dans le monde.

Geensys offre un éventail de services et de produits de qualité dont les synergies s'affirment sur l'engagement de l'entreprise à fournir à ses clients des solutions globales, proches de leurs métiers et respectueuses de leurs préoccupations économiques, dans les secteurs de l'automobile, l'aéronautique, la défense, le ferroviaire, l'industrie, le médical et les télécommunications.

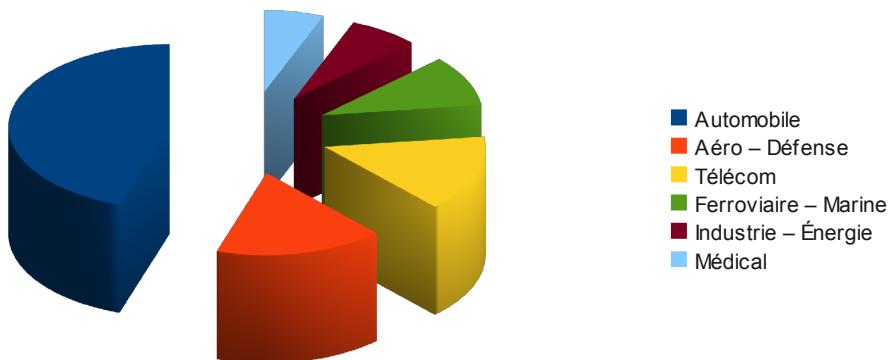


Illustration 2: Part de l'activité de Geensys par domaine

En résumé, la société est une entreprise innovante dans de nombreux secteurs. Le bilan de l'entreprise fait preuve de sa vitalité et de son engagement dans la recherche au vue des éléments suivants :

Chiffre d'affaires	22 Millions d'€
R&D	22% du CA
Export	18% du CA
Effectif	245 personnes

Tableau 2: Chiffres clés en 2008 pour Geensys

Historique

Geensys est issue de la fusion en 2007 de deux sociétés, Ayrton Technology et TNI Software. La société est ainsi une nouvelle entreprise, entièrement dédiée à l'embarqué, plus proche du métier et des préoccupations de ses clients, plus apte à les accompagner par un centrage sectoriel fort, une expertise méthodologique transverse ainsi que des capacités d'exécution plurielles. La société se prépare à prendre une place de premier plan sur son marché en créant une nouvelle force centrée sur les projets de systèmes, matériels et logiciels embarqués de ses clients.



Illustration 3: Logo d'Ayrton Technology

Ayrton Technology

Depuis sa création, Ayrton Technology s'est affirmée comme l'une des jeunes sociétés françaises les plus remarquables par la dynamique soutenue de sa croissance et le positionnement distinctif de ses produits de service pour l'embarqué, dans les secteurs des Transports, de la Défense, des Télécommunications et de l'Industrie.

En 2005, l'entreprise devient la plus jeune et troisième société du secteur à obtenir une certification CMMI de niveau 3. Sa croissance soutenue est couronnée au plan national par le Prix Gazelle.

Elle s'engage en 2006 aux côtés de TNI-Software dans O4A l'un des projets d'innovation du Pôle Automobile Haut de Gamme les plus saillants du secteur de l'embarqué automobile, centré sur le standard d'architecture logicielle AUTOSAR.

En Janvier 2007, l'entreprise inaugure l'extension de son Laboratoire d'ingénierie sur 1000m² à Carquefou. Le 22 juin 2007, de sa fusion avec TNI-Software, éditeur français de logiciels scientifiques pour les systèmes embarqués de notoriété internationale, elle crée Geensys S.A.S .



Illustration 4: Logo de TNI Software

TNI Software

Depuis sa transformation et son recentrage sur le développement d'outils et le support méthodologique dédiés au secteur des systèmes embarqués à forte dominante logicielle, TNI-Software a fait preuve d'une remarquable dynamique sur son marché, à plusieurs titres :

En 2005, TNI-Software devient Premium Member du consortium AUTOSAR et annonce sa plateforme d'outils AUTOSAR basée sur les technologies Open Source de la Fondation Eclipse. Cette même année, elle ouvre son premier bureau en Chine (Shangaï).

Dans le secteur de l'aéronautique et de l'espace, la société se distingue par sa contribution à la création du User Group DO254, dont elle est aujourd'hui le co-administrateur.

En collaboration avec PSA, elle crée le Pôle Automobile Haut de Gamme et prend l'initiative du projet O4A, l'un des projets de pôles les plus avancés sur la nouvelle architecture logicielle du standard AUTOSAR, en cours d'adoption par une très grande majorité des acteurs du secteur électronique automobile au plan mondial.

En 2006, TNI-Software annonce une nouvelle version de son atelier de contrôle commande ControlBuild, qui est la première solution du marché, basée sur des modèles (Model Based Design), à automatiser la totalité du cycle de conception système. Plus tard dans l'année, la société ouvre son bureau au Japon à Yokohama où elle démarre avec succès la commercialisation de Reqtify sur le secteur automobile. Elle signe ensuite, pour son produit phare Reqtify, de multiples accords de distribution et d'OEMs avec des partenaires de dimension internationale (Artisan, I-Logix / Telelogic, National Instruments, Esterel Technology, LDRA, ...).

Membre Add-in Provider de la fondation Eclipse, elle devient très rapidement un acteur majeur du développement de l'écosystème Eclipse en Europe et en France sur le secteur de l'embarqué.

Organisation actuelle de la société

La société est actuellement organisée suivant 2 axes, par 'Business Unit' et par 'Business Line'. Chaque 'Business Unit' et 'Business Line' est sous la responsabilité unique d'une personne pour le groupe. L'ensemble des activités est aussi contrôlé par des moyens transverses comme le service Qualité, la gestion des Ressources Humaines, le service Commercial et la Comptabilité.

Chaque 'Business Unit' représente un domaine d'activités de l'entreprise et possède des compétences à la fois dans le monde du système débarqué et du système embarqué. Ce concept a ensuite été étendu à l'ensemble des activités réalisées dans un pays en particulier ou une zone géographique.

Chaque 'Business Line' représente un axe de développement de Geensys. On peut citer notamment deux pôles importants, la prestation de services et la réalisation d'études.

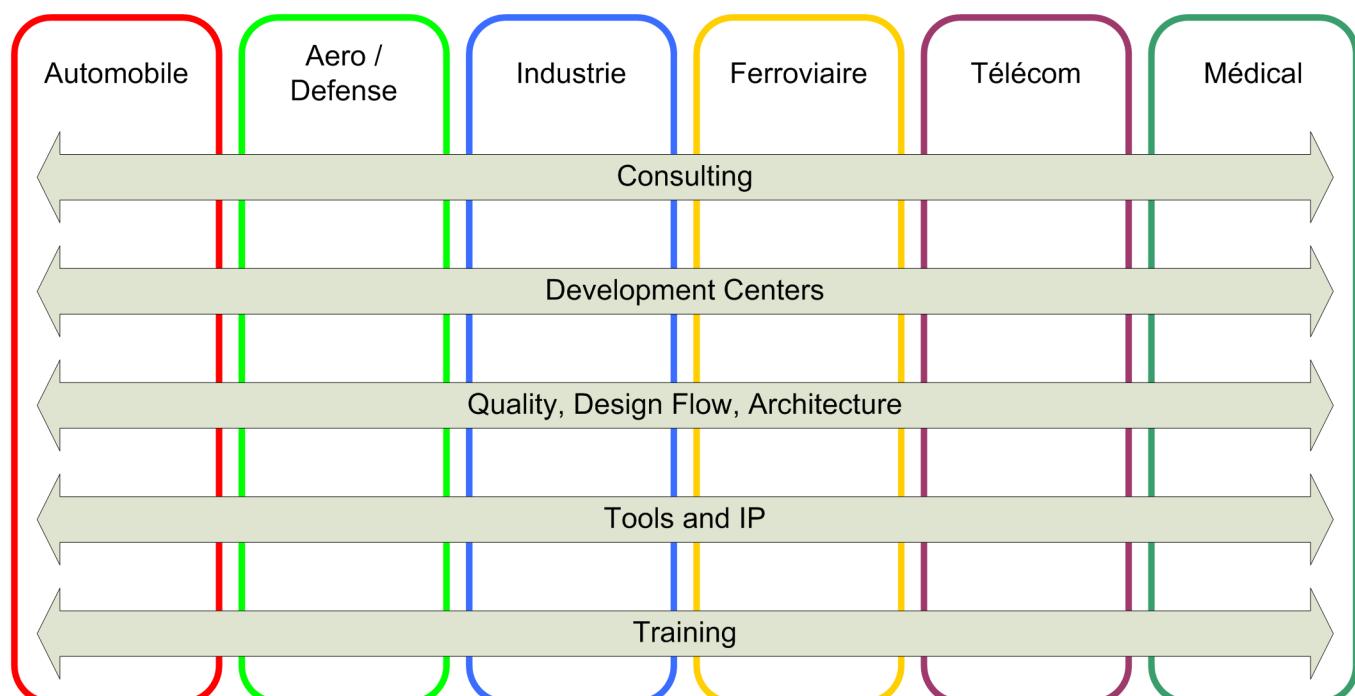


Illustration 5: Organisation et responsabilités dans l'entreprise

Partenaires et Références

La société est membre de plusieurs pôles de compétitivités notamment des pôles ID4CAR et mov'eo dans le domaine automobile. Une collaboration étroite avec divers partenaires dans le cadre de consortiums internationaux a aussi été mise en place comme dans le consortium AUTOSAR ou dans la fondation Eclipse.

Geensys travaille aussi en partenariat avec des intégrateurs de solutions métiers afin de placer ses produits dans une solution complète. On peut citer notamment les solutions SysML d'Artisan Software et Matlab de MathWorks qui collaborent avec l'outil Reqtify de Geensys.

Automobile	Aéronautique	Ferroviaire	Industrie	Télécoms	Médical

Tableau 3: Références par 'Business Unit'

Geensys à Carquefou



Illustration 6: Photo du site de Carquefou

L'ancien site d'Ayrton Technologies m'a accueilli tout au long de mon stage. Les locaux sont situés dans la zone d'activités La Fleuriaye à Carquefou. Sur ce site, la société possède essentiellement des compétences en systèmes embarqués.

Dans les différents projets menés actuellement, on peut citer :

- O4A - phase II : ce projet réalisé dans le cadre du pôle ID4CAR consiste à définir un ensemble de méthodes et d'outils pour implémenter une solution AUTOSAR.
- ARPOD : il s'agit d'une réponse à un besoin de solution et de formation à la société Autocruise concernant le standard AUTOSAR.
- Actaris : c'est une activité plateau pour le compte d'EDF. Le développement d'une nouvelle génération de compteur électrique y est étudié.
- PL7 : le développement d'un tableau de bord pour automobile à la demande d'un équipementier de rang 1.

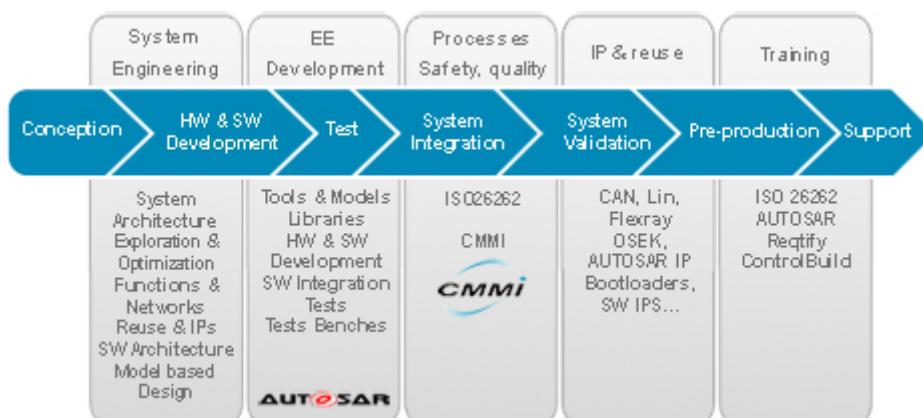


Illustration 7: Flux de compétences pour l'automobile

Géographiquement dispersé entre les sites, le projet de plate-forme AUTOSAR est une réalisation qui n'aurait pas été possible sans la fusion des deux entreprises. Il est situé dans la 'Business Unit' automobile et utilise l'ensemble des 'Business Line'. Le département AUTOSAR situé à Carquefou s'occupe du développement des couches basses et de l'offre de formation dans le cadre du projet O4A.

- *Ingénierie Système* : Geensys-AUTOSAR est un ensemble logiciel embarqué, outil de génération, outil de simulation qui démontre le savoir faire de l'entreprise dans une architecture logicielle complexe.
- *Centre de Développement* : le développement des modules embarqués est soutenu par les différents pôles de compétitivités à vocation automobile.
- *Processus Métiers* : la solution répond aux dernières exigences de qualité des normes automobiles.
- *Outils et Propriété Intellectuelle* : la solution peut-être fournie sous différentes formes, notamment comme application de démonstration de l'outil AUTOSAR Builder ou de solution clé en main pour réseaux CAN, FlexRay et LIN.
- *Formation* : la volonté de la société à s'implanter durablement dans le domaine automobile à conduit à la création d'une offre de formation AUTOSAR dans le cadre des pôles de compétitivité.

Contexte du stage

Dans un premier temps, mon activité de stage s'est déroulé dans le cadre d'un projet de recherche O4A phase II puis pour le projet ARPOD. Le projet O4A est une projet de recherche financé dans le cadre du pôle ID4CAR (anciennement Automobile Haut de Gamme).



Illustration 8: Logo du pôle ID4CAR

Ce projet vise à implémenter une architecture AUTOSAR de référence qui permettra de valider les nouveaux concepts du monde automobile. Dans ce cadre, l'ajout du réseau Ethernet permet à cette architecture innovante de répondre à de nouveaux besoins comme la mise à jour en fin de chaîne ou le téléchargement de nombreuses informations de développement.

Le projet ARPOD utilise cette architecture générique afin de développer rapidement des contrôleurs embarqués. Le besoin d'un réseau haut-débit c'est fait ressentir dans ce projet et la solution Geensys/AUTOSAR accompagné de la pile TCP/IP y répondait.

Le consortium AUTOSAR

AUTOSAR est une architecture logicielle standard et ouverte pour l'automobile. Elle a été conjointement développée par des constructeurs automobiles, des sous-traitants et des fournisseurs d'outils.



Illustration 9: Logo du consortium

Les buts de ce standard sont d'ouvrir la voie à l'innovation dans les systèmes électroniques pour plus de performances, de sûreté de fonctionnement et maîtrise de consommation. Le consortium a aussi pour but d'établir un partenariat fort qui vise à « coopérer pour le standard et rivaliser pour l'implémentation ».

Status: 13th March 2008

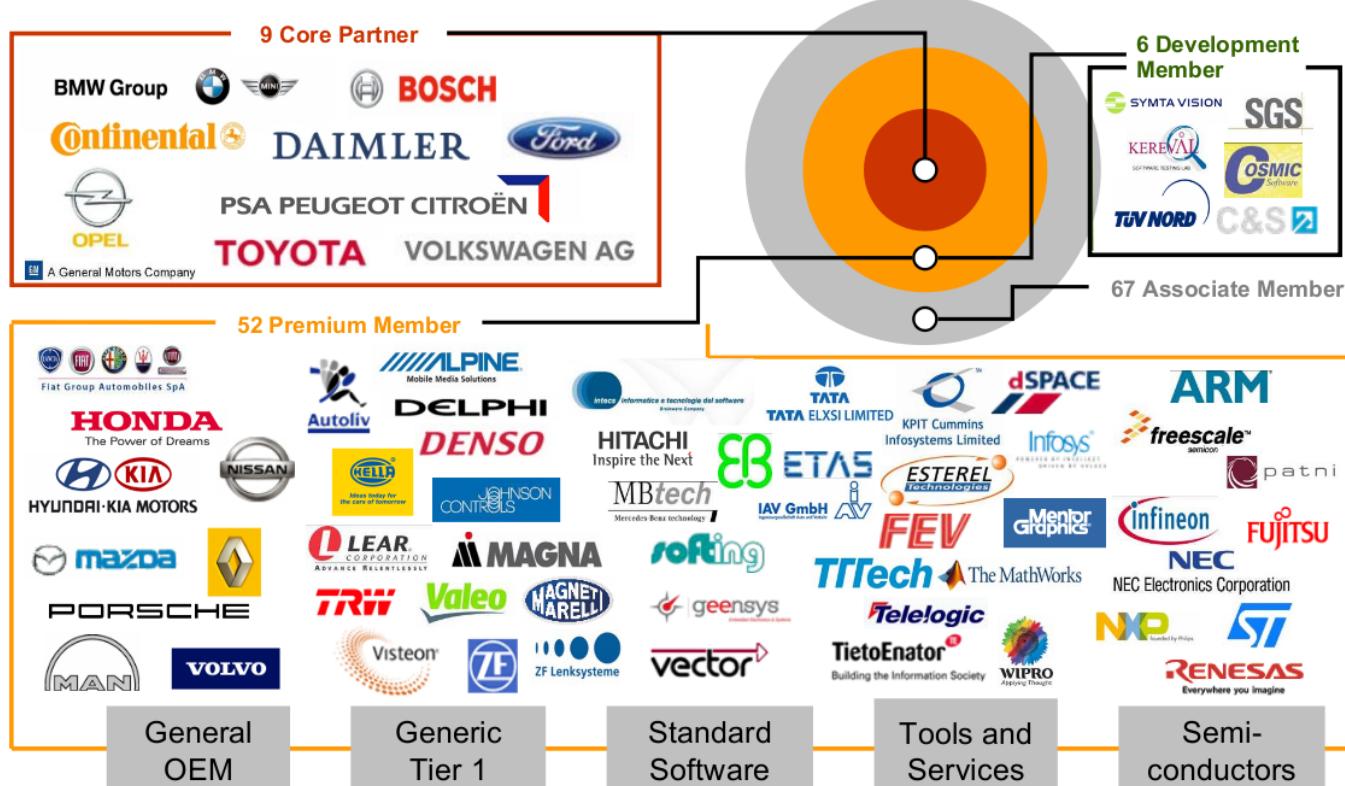


Illustration 10: Membres du consortium AUTOSAR par niveaux de participation et compétences.

D'un point de vue technique, la standardisation permet de gérer la complexité grandissante des systèmes électroniques. Elle permet de préparer l'arrivée de futures technologies et de gérer les coûts d'efficiencies sans diminuer la qualité. Cela permet aussi de faciliter les échanges et la mise à jour des logiciels et matériels tout au long de la vie du véhicule.

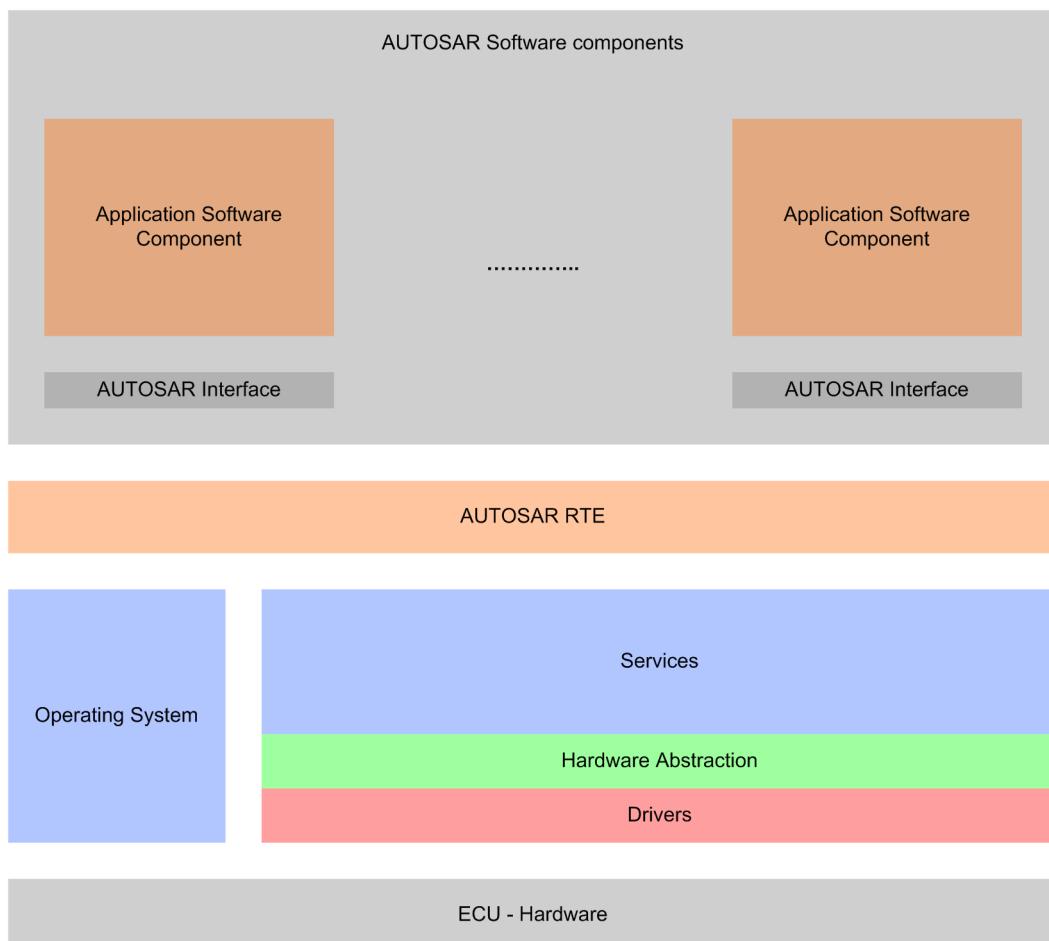


Illustration 11: Architecture simplifiée d'un ECU AUTOSAR

La configuration statique de l'ensemble des modules permet de réaliser une plate-forme générique tout en gardant une empreinte mémoire réduite. L'architecture AUTOSAR généralise aussi des techniques qui permettent de configurer les différents emplacements mémoire de manière fine par module. La responsabilité d'un bug est ainsi clairement établie et facilement corrigable même dans une configuration statique.

Afin de gérer cette configuration, des outils sont aussi indispensables.

Objectif final du stage

L'objectif final de ce stage est d'apporter à l'équipe AUTOSAR chez Geensys une compétence Ethernet dans le cadre d'AUTOSAR. À noter que les principaux concurrents de cette équipe, Vektor et ElectroBit, proposent déjà ou travaillent sur des solutions équivalentes pour leurs clients.

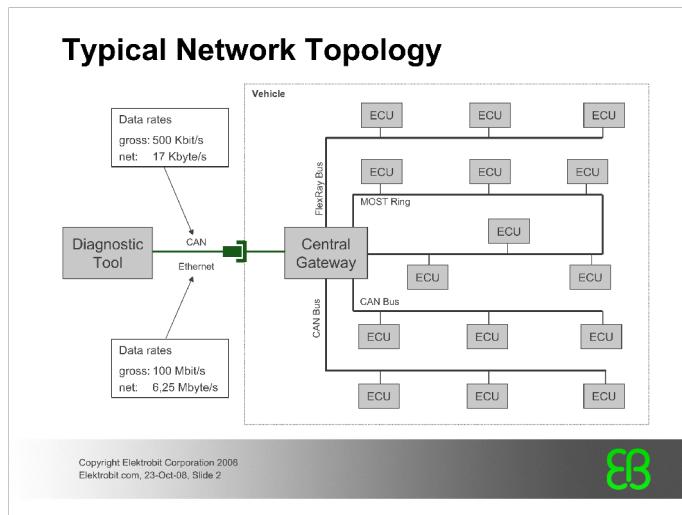


Illustration 12: Extrait de la documentation ElektroBit/AUTOSAR supportant Ethernet.

Au cours de ce stage et dans le cadre d'un projet client, j'ai dû aussi accompagner l'équipe dans la maîtrise de la solution développée. L'intégration devait ainsi être réalisée en binôme et expliquée pas à pas, afin de garantir la transmission des connaissances.

Au final, l'ensemble de la solution Geensys devra être intégrée comme une application de démonstration de la carte à base de MPC5668.

Contraintes

Les spécifications AUTOSAR étant classifiées comme 'Draft' (ébauche), il était nécessaire de suivre celles-ci au maximum tout en gardant une distance d'appréciation sur des éléments non figés. De plus, les problèmes rencontrés lors du développement peuvent être dûs à une mauvaise spécification. Dans ce cas, ils devront être remontés au consortium AUTOSAR.

L'ensemble de développement sera aussi réalisé suivant les conventions et la méthodologie Geensys pour AUTOSAR. Aussi, les exigences de qualité logicielles de l'automobile devront être appliquées. La configuration devra aussi être intégré à l'outil comme tout module de production.

Toute activité sera justifiée dans un planning global d'activités afin d'extraire des métriques applicables sur les prochaines intégrations. Chaque activité est ainsi planifiée à priori et à posteriori.

Équipe en charge de l'intégration

J'ai tout d'abord intégré une première équipe dans le cadre de la solution sur la carte d'évaluation pour AVR32. Cette équipe est composée des personnes suivantes :

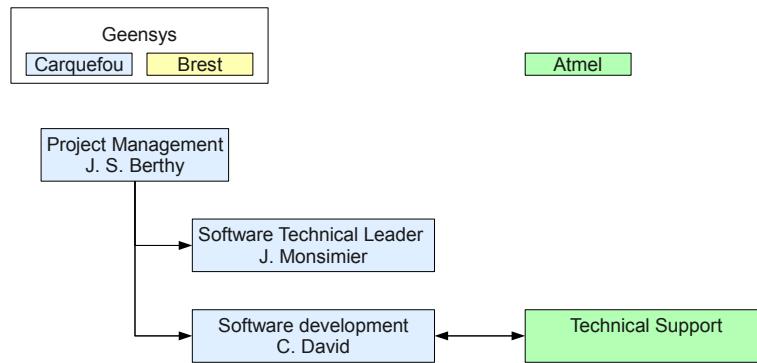


Illustration 13: Organigramme de la solution Ethernet-AVR32

- Jean Sébastien Berthy pour la gestion de projets et de personnes.
- Julien Monsimier pour le suivi au quotidien et comme référent technique sur AUTOSAR.
- Clément DAVID pour le développement logiciel, l'intégration et la documentation.

Dans le cadre du projet ARPOD, l'organisation est plus complète. L'organigramme n'est pas celui du projet complet mais un résumé centré sur l'activité de Geensys - Carquefou.

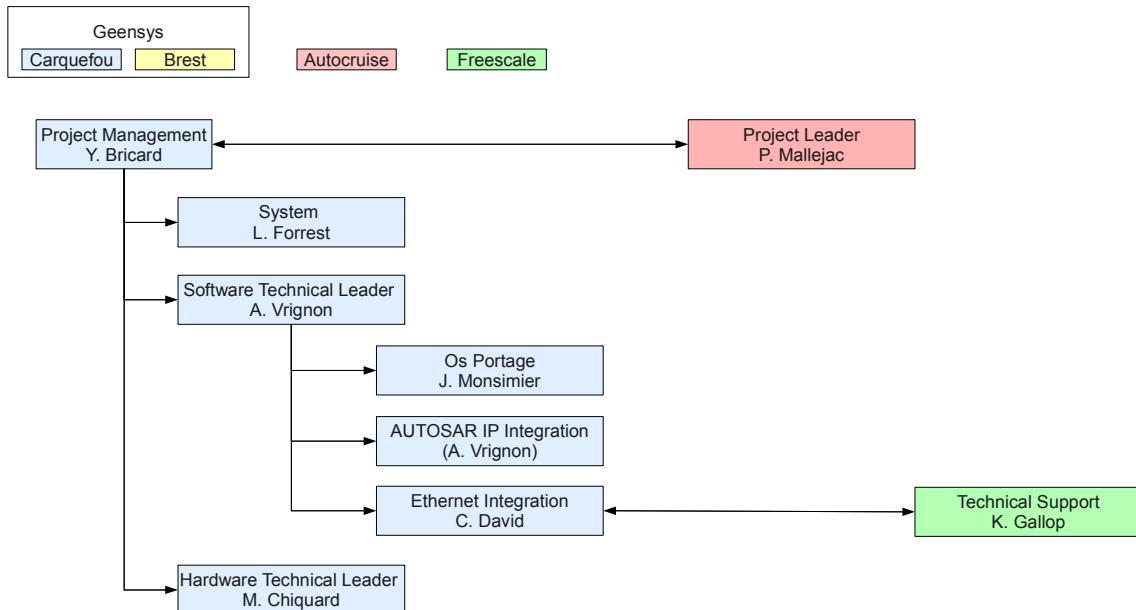


Illustration 14: Organigramme de l'intégration pour ARPOD

Les activités de gestion de projet, développement et validation sont ainsi totalement réalisées sur le site de Carquefou. Un contact avec le responsable de projet chez AutoCruise et le support technique de chez Freescale m'a aussi permis de résoudre des problèmes au niveau pilote.

Planning initial

Le planning établi dans les premières semaines du projet tient uniquement compte de la démonstration sur cible AVR32. En effet, à cette date, le projet client n'était pas connu donc non planifié dans mon activité. Le planning initial complet peut être trouvé en annexe et seul un extrait des tâches principales figure ici.

Le développement a été initialement découpé en trois tâches principales elles-mêmes redécoupées en sous-tâches. Le listing ci-dessous donne une description succincte du travail à réaliser et de la durée estimée.

- Travail sur la pile TCP/IP (65 jours)
- Implémentation du module AUTOSAR permettant l'intégration de la pile avec la messagerie. Le module a été développé suivant la méthode Geensys CMMI3. (35 jours)
- Intégration du Kit sur AVR32 (44 jours)

Ce planning initial a du évoluer dès les premières semaines de stage suite aux contraintes horaires du projet ARPOD. Une première intégration a ainsi eu lieu sur cible AVR32 mais rapidement une cible PowerPC MPC5668 a été préférée pour l'intégration finale. Le projet a ensuite été échelonné suivant trois jalons, le planning effectif complet peut-être trouvé en annexe de ce document.

- intégration/maîtrise de la pile TCP/IP sur AVR32.
- Ajout du module supplémentaire pour une validation des principes d'AUTOSAR sur AVR32.
- Portage de l'ensemble sur cible MPC5668.

Travail effectué

Un premier travail a consisté à identifier rapidement les exigences du document de spécification afin de choisir un composant TCP/IP remplissant les besoins de la spécification.

La spécification étant dans un stade de brouillons, il m'a fallu, tout au long de mon stage, remettre en question les exigences formulées et remonter au consortium les erreurs. Ce travail m'a permis d'avoir une vision plus globale de l'architecture et de découvrir en profondeur les concepts d'AUTOSAR.

À la date de planification, la solution Geensys/AUTOSAR supportant l'Ethernet devait être incorporée dans la version suivante, 4.0, de l'architecture. Cependant pour le projet ARPOD, il a aussi fallut l'intégrer dans une architecture 3.1 pour bénéficier des modules déjà validés pour cette version.

Critères de choix de la pile TCP/IP

Avant de commencer le travail d'intégration, j'ai d'abord du sélectionner une pile TCP/IP qui correspondait au besoin de l'application. Tout d'abord, le choix de la licence s'est rapidement imposée. Une licence sans obligation de redistribution était en effet nécessaire afin de correspondre aux différentes formes de livraisons de la solution.

Synthèse de la spécification

Le monde de l'automobile a défini ses propres moyens de communication répondant à ses propres contraintes depuis la fin des années 90. Cependant, le besoin naissant d'un réseau haut débit commence à se faire ressentir pour répondre à plusieurs demandes des constructeurs (téléchargement de code, diagnostique des fonctions électroniques).

Le consortium AUTOSAR a travaillé sur la spécification d'une pile TCP/IP intégrée comme un COTS³. Compte tenu de ce choix, de nombreuses variantes ont été acceptées afin de satisfaire l'ensemble des acteurs du consortium, ce qui rend la spécification du module TcpIp partielle. L'interface supérieure du module est basée sur le standard POSIX 1003.1 (socket) alors que l'interface inférieure est basée sur l'expérience du consortium (quasi-identique aux modules Can, Lin et FlexRay).

Afin de faire la relation entre l'interface supérieure et le module de communication générale (PDU Router⁴), il a fallu créer un nouveau module qui transforme un flux de données en des signaux asynchrones. Ce nouveau module a été nommé 'Socket Adaptor' et est entièrement configurable et spécifié comme tout autre module AUTOSAR.

³ Commercial Off The Shelf : Solution sur étagère

⁴ Protocol Data Unit Router : Réalise la liaison entre la couche de communication applicative et les différents réseaux

Plusieurs autres modules ont été spécifiés par le consortium afin d'avoir une cohérence entre les différents moyens physiques (Can, Lin, etc...). On peut citer notamment les modules EthNM⁵ et EthSM⁶ qui s'occupent de gérer les phases de vie du réseau et de diagnostiquer l'état du lien physique.

Une pile qui cible l'embarqué.

Tout d'abord, une pile TCP/IP a dû être sélectionnée afin de répondre à cette demande. Cette pile doit bien entendu cibler les architectures embarquées mais aussi être portable sur plusieurs systèmes d'exploitations (notamment OSEK/AUTOSAR) et facilement configurable pour entrer dans le cadre d'AUTOSAR.

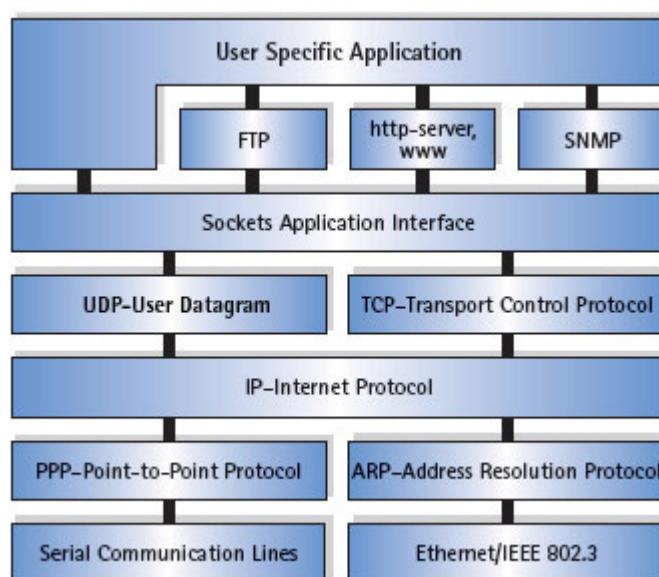


Illustration 15: Diagramme d'une pile TCP/IP générique

De ce constat, on peut éliminer les piles spécifiques à un système d'exploitation (Linux, ...). Les piles propriétaires ne sont pas intéressantes pour l'entreprise car elles ont peu de valeur ajoutée. Les piles '*BSD', bien que relativement portables (notamment sur des systèmes temps réels), ne sont configurables qu'à l'exécution et non avant.

⁵ Ethernet Network Manager : voir p.27 : Architecture choisie

⁶ Ethernet State Manager : void p.27 : Architecture choisie

Après une veille technologique sur les piles libres, deux ont pu être sélectionnées car elles répondaient aux besoins énoncés ci-dessus :

- lwIP

Pile open-source ciblant les processeurs 16bits et 32bits. Elle est portable sur les systèmes d'exploitations préemptifs et non-préemptifs.

- uIP

Pile open-source du même auteur que lwIP mais ciblant les processeurs 8bits et portable sur les systèmes d'exploitations préemptifs et non-préemptifs.

Contraintes supplémentaires

D'autres contraintes doivent aussi être prises en compte dans la sélection d'une pile TCP/IP notamment sa conception au niveau de la gestion de la mémoire, du support de plusieurs interfaces ou des différents protocoles supportés (DNS⁷, IGMP⁸, SNMP⁹, etc...).

⁷ Domain Name Service : Protocole qui permet d'associer un nom à chaque composant du réseau.

⁸ Internet Group Management Protocol : Permet de réaliser du multicast sur réseau IP.

⁹ Simple Network Management Protocol : Outils d'administration des noeuds du réseau, permet de diagnostiquer et configurer les équipements du réseau.

Architecture choisie

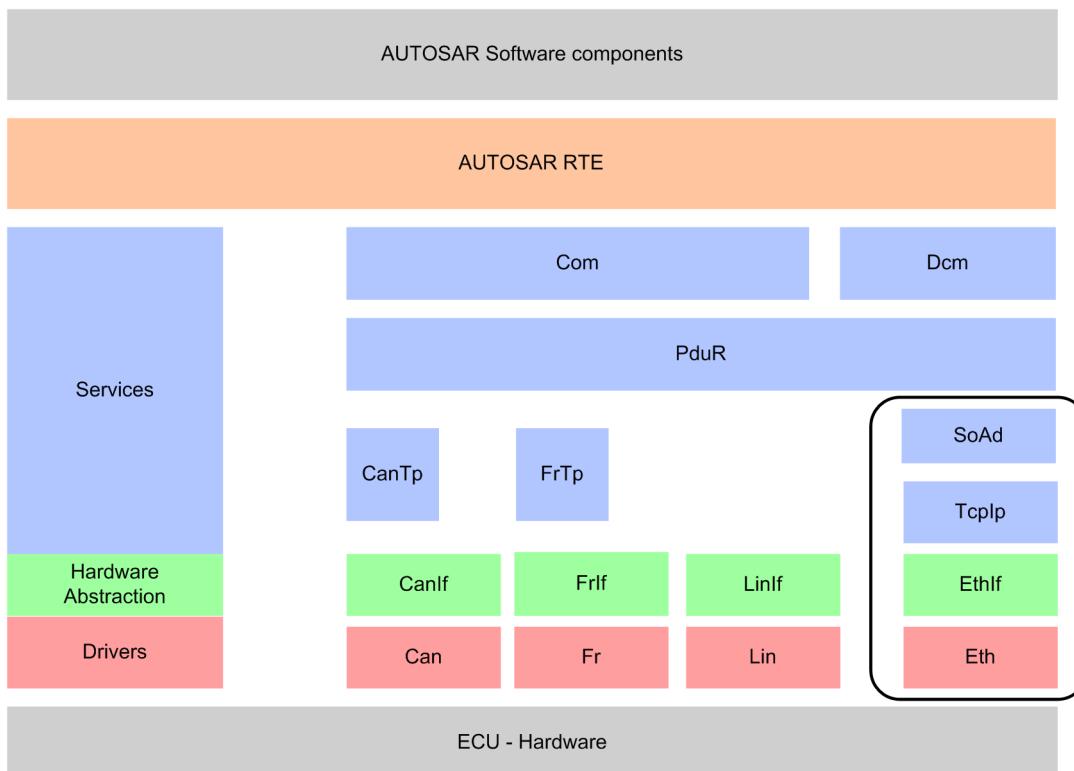


Illustration 16: Architecture des couches basses AUTOSAR supportant TCP/IP

Les différents modules qui doivent être obligatoirement implémentés sont:

- Le module Tcplp (TCP/IP) : ce module implémente les machines à état du protocole TCP, celle du protocole UDP et la gestion de l'IP. D'autres protocoles sont aussi implémentés ainsi qu'une gestion de la mémoire à travers un espace réservé à l'allocation.
- Le module SoAd (Socket Adaptor) : ce module réalise la transformation entre la communication par message tel qu'implémenté dans AUTOSAR et la communication en flux comme dans le protocole TCP.
- Le module EthDrv (Ethernet Driver) : ce module est l'équivalent des drivers fournis par chaque constructeur pour une carte donnée. Il doivent être adaptés pour AUTOSAR.

D'autres modules optionnels sont spécifiés par le consortium, ils ne sont pas obligatoires et peuvent donc être planifiés sans contraintes. On peut citer notamment :

- Le module EthNM (Ethernet Network Manager) : gère les phases de vie de la communication.
- Le module EthSM (Ethernet State Manager) : gère les phases de vie du driver.
- Le module EthIf (Ethernet Interface) : lorsqu'une carte possède plusieurs interfaces physiques, il réalise le lien entre un message et son interface correspondante.

La spécification AUTOSAR autorise plusieurs variantes dans l'implémentation. Nous avons choisi d'implémenter une variante simple afin d'avoir un démonstrateur rapidement opérationnel. Dans cette variante, la pile TCP/IP est intégrée en même temps que le driver Ethernet. Dans un souci d'évolution de cette architecture, les modules manquants sont d'ores et déjà planifiés.

Le module TCP/IP (Tcplp)

Le composant logiciel sélectionné est la pile TCP/IP lwIP. Il a été choisi car il correspondait mieux aux spécifications du consortium. En effet, certains membres du consortium ont formulé l'exigence d'avoir les fonctionnalités de l'algorithme de Nagle¹⁰, ce que ne supporte la pile uIP.

¹⁰ L'algorithme de Nagle permet de diminuer la bande passante occupée par les entêtes TCP en retardant l'émission de données utiles.



Illustration 17: Logo du projet lwIP

De plus, d'un point de vue opérationnel, les différentes cibles AUTOSAR sont des micro-contrôleur 16bits ou 32 bits ce qui correspond à la conception de la pile lwIP.

La pile lwIP est décomposé en modules logiciel implémentant chacun une fonctionnalité propre. Certains s'occupent de l'implémentation de la gestion des protocoles (Tcp, Udp, Ip, Icmp, ...), d'autres de l'abstraction au matériel et enfin d'autres de l'ajout de fonctionnalité génériques comme la gestion mémoire.

Cette pile TCP/IP est aussi facilement configurable. En effet, l'utilisation poussée du pré-compilateur permet d'activer ou de désactiver des fonctionnalités facilement. Cette possibilité est très utilisée dans l'architecture AUTOSAR à travers les outils de configurations, ce qui facilite l'intégration.

D'un point de vue extérieur, on obtient donc la vue ci-dessous, suivant la méthode MCSE¹¹.

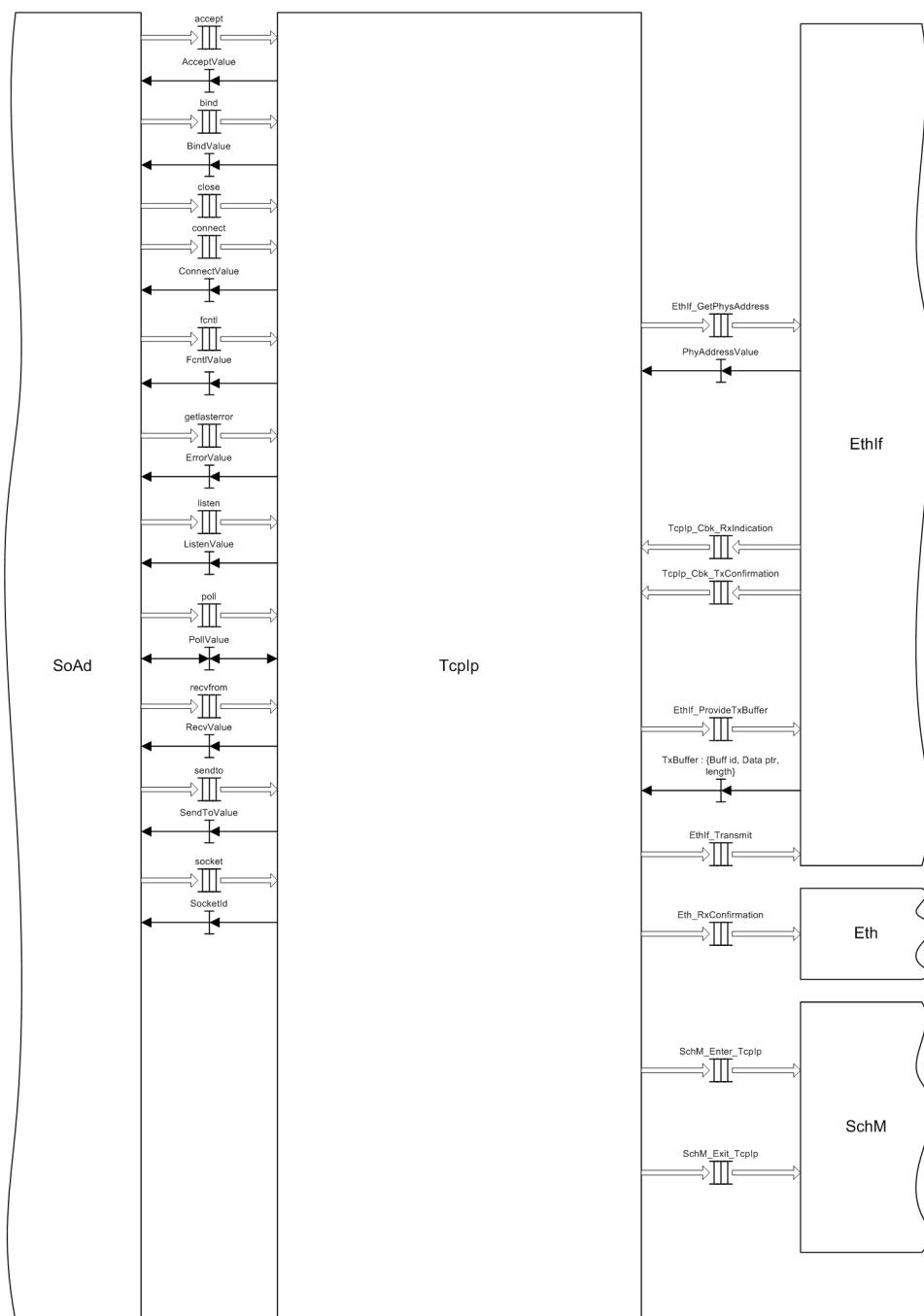


Illustration 18: Vue externe du module TcpIp

¹¹ La méthode MCSE (Méthodologie de Conception des Systèmes Électroniques) est une méthode de conception conjointe développée par Jean-Paul Calvez à l'Université Polytechnique de Nantes.

Comme souligné précédemment, le découpage en fonctionnalités conduit à une pile TCP/IP complète. On remarque, ci-dessous, que différents composants garantissent un portage aisément de la pile. On peut donc facilement cibler les problèmes potentiels. Ceux-ci sont principalement situés dans les couches de gestion de la mémoire et abstraction de la pile.

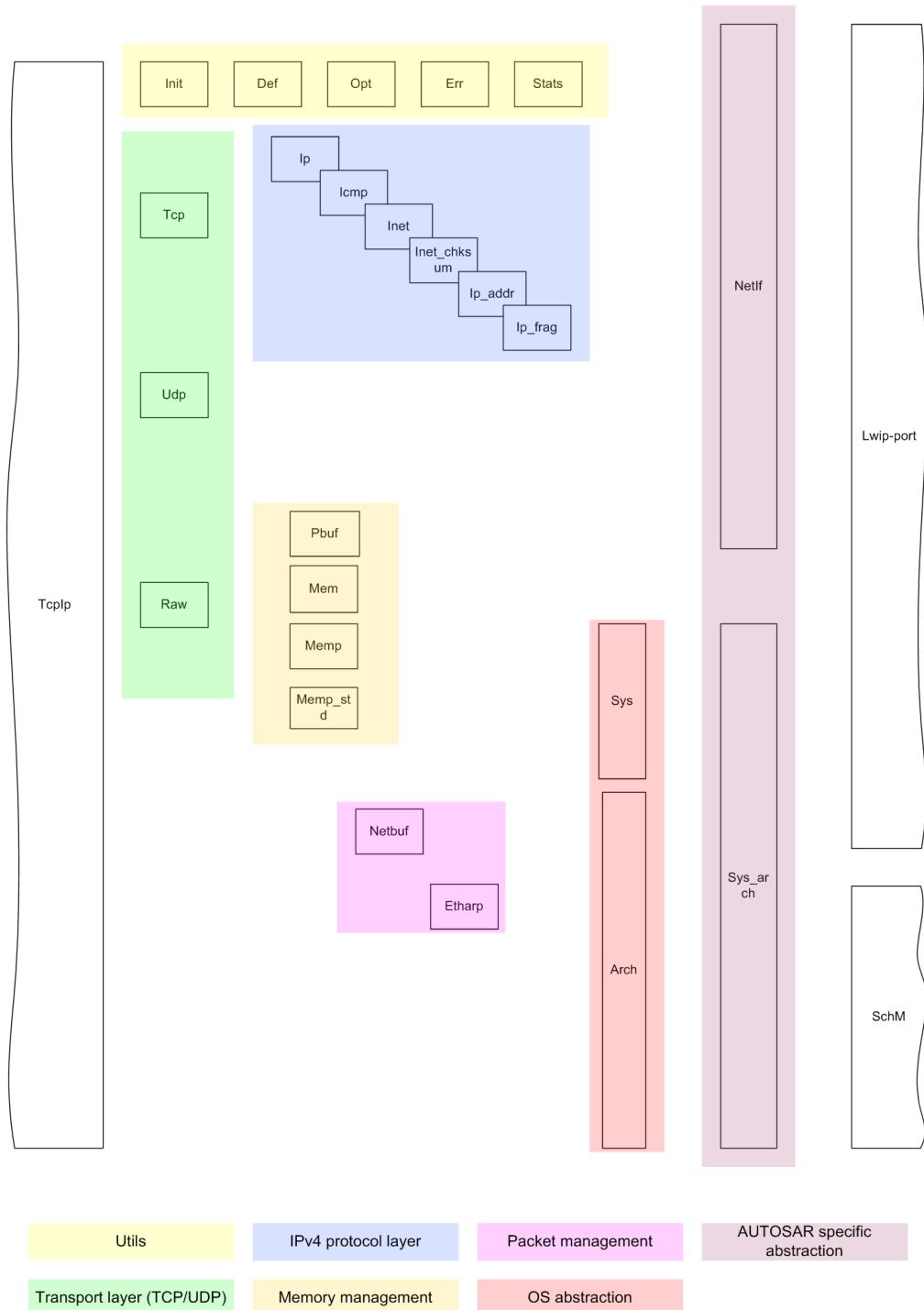


Illustration 19: Décomposition en fichiers de la pile lwIP

La configuration de la pile est réalisée durant la phase de pré-compilation mais certains paramètres nécessitent d'être figés par des choix de conception. Par exemple : le fichier socket.c implémente l'API POSIX 1003.1 qui est ré-implémenté dans le module afin de répondre aux contraintes de spécification (API non-bloquante uniquement) et d'évolution (ajout d'une API AUTOSAR).

Le module Socket Adaptor (SoAd)

Le module SoAd permet de réaliser la transformation de données entre la pile TCP/IP qui utilise des flux de communications à travers l'API POSIX 1003.1 et le mécanisme de communication d'AUTOSAR qui utilise des messages appelés PDU¹². Ce module permet aussi de configurer les différents flux de communication utilisés et permet donc de rejeter les flux non-configurés.

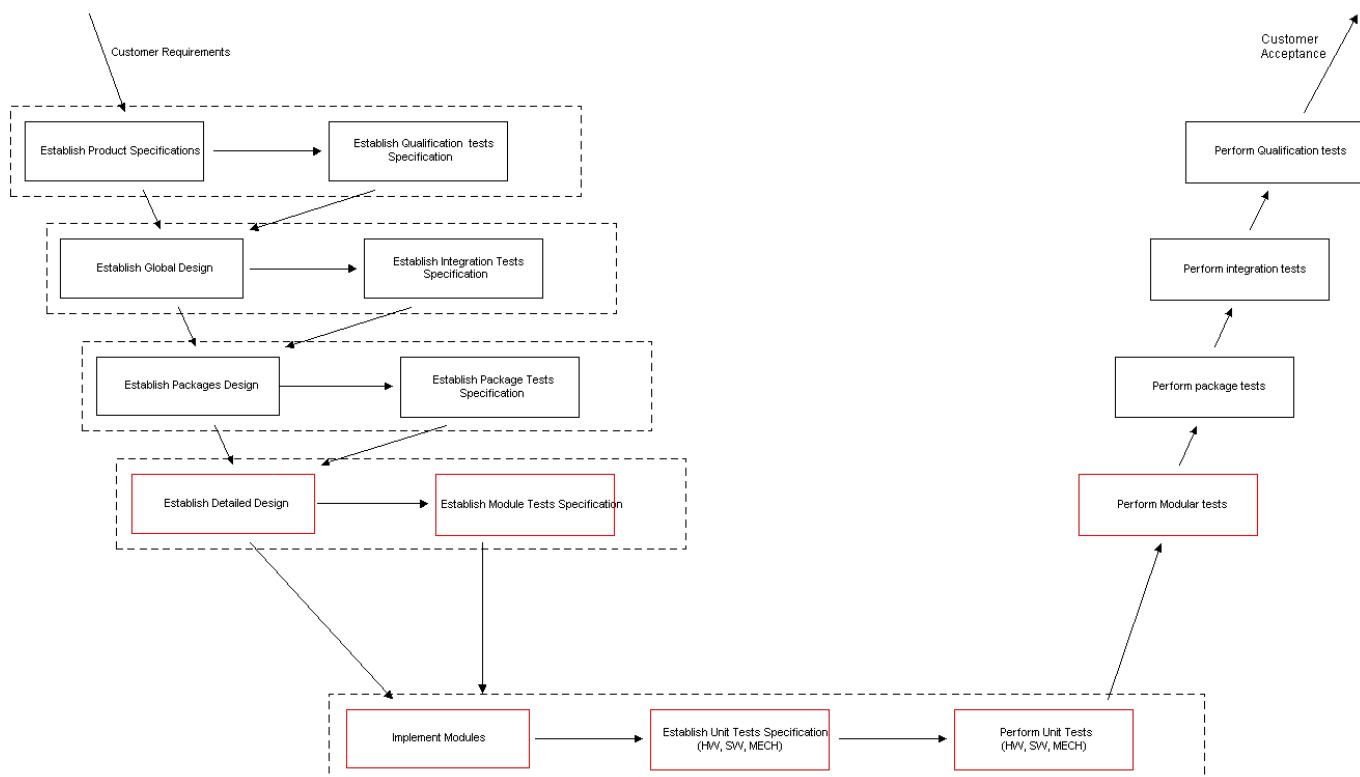


Illustration 20: Couverture du cycle en V lors de la planification d'un module.

¹² Protocol Data Unit : Message contenant des données et leur taille.

Ce module est spécifique à AUTOSAR et est implémenté suivant la méthode MCSE/Geensys pour AUTOSAR. Le 'cycle en V' a donc été parcouru sur les parties conception, implémentation et validation d'un module.

Le module Ethernet Driver (EthDrv)

Le module EthDrv a été implémenté plusieurs fois tout au long de ce stage. Tout d'abord une première fois en utilisant le pilote fourni par le fabricant Atmel et en le travaillant pour le rendre utilisable avec la pile TCP/IP. Une seconde fois, ré-écrit complètement afin de dissocier le driver de la pile. Une troisième fois ré-écrit complètement pour la cible du projet ARPOD. Et enfin, une quatrième fois en intégrant le module générique EthIf. Cette dernière intégration n'était pas obligatoire mais a permis à Geensys de former des personnes au Vietnam (Gate Technologies) au développement de modules AUTOSAR.

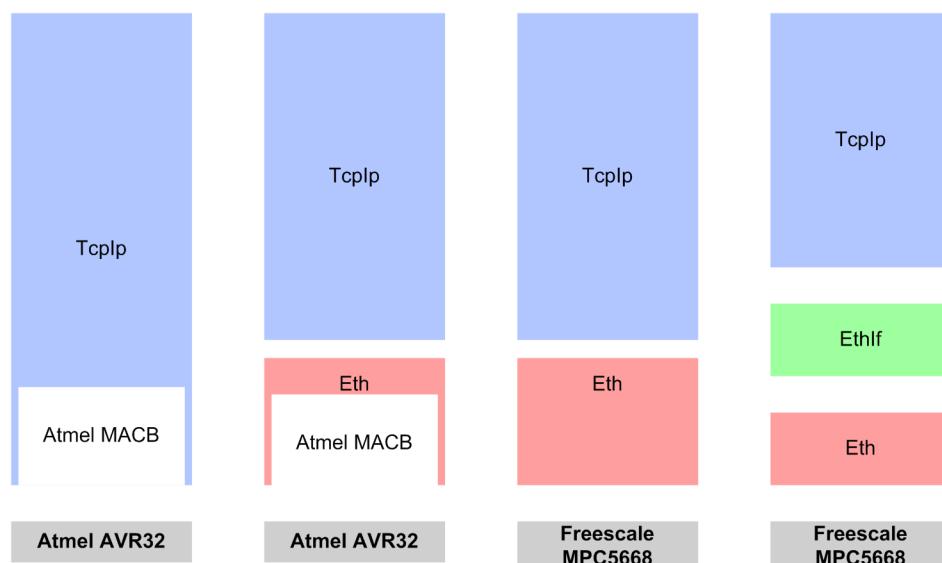


Illustration 21: Les différentes intégrations du driver Ethernet.

Critiques de cette architecture

En voulant intégrer une pile TCP/IP comme composant COTS, le consortium a privilégié une intégration partielle dans les principes d'AUTOSAR. Ainsi, l'interface de programmation utilisée fonctionne d'une façon très dynamique et non configurable. Une ré-implémentation future d'une pile TCP/IP permettra de réduire la taille de code en figeant l'ensemble des options à la configuration.

On notera aussi une copie mémoire obligatoire au niveau de la couche basse (dans la relation entre les modules SoAd et TcpIp) ce qui diminue les performances globales du système et peut casser le fonctionnement temps-réel d'une application (copie en interruptions bloquées). Cette obligation est dûe à la réutilisation de l'interface de programmation standard. Bien qu'une interface de programmation différente et ne nécessitant ni copie, ni modification des concepts AUTOSAR ait été spécifiée, celle-ci est partielle et non-fonctionnelle en l'état.

Les différents diagrammes de séquences concernant l'interaction entre les modules sont fournis en annexe de ce document.

L'application de démonstration



Illustration 22: Logo de la société Autocruise, projet ARPOD

L'application décrite ici est issue du projet ARPOD. Ce projet financé dans le cadre du pôle ID4CAR a pour but de « Développer un pare choc intelligent, intégrant des capteurs miniaturisés, associés à une unité centrale de traitement. L'objectif est d'améliorer la perception périphérique de l'environnement du véhicule pour un meilleur contrôle du véhicule. » (site du pôle ID4CAR).

Le projet ARPOD a imposé une architecture distribuée nécessitant un réseau à haut débit. Ceci afin de transmettre une grande quantité d'informations en un temps réduit. La solution Geensys/AUTOSAR a été sélectionnée sur l'ensemble des composants du système.

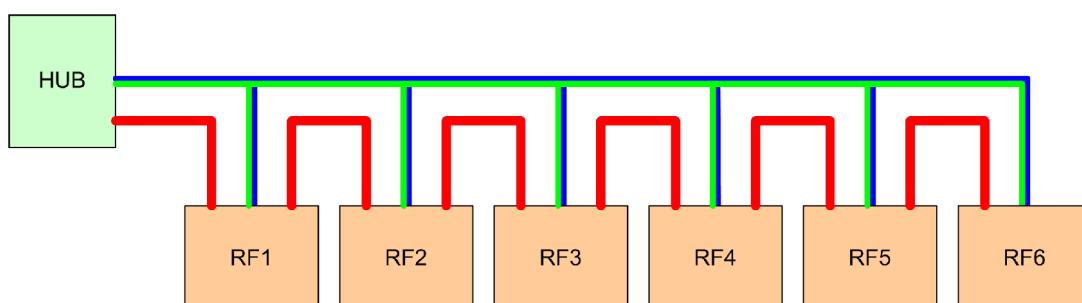


Illustration 23: Topologie réseau utilisée dans le projet ARPOD. Le réseau Ethernet est ici représenté en rouge, le réseau CAN en vert et l'alimentation en bleu.

Le projet est ainsi décomposé en six contrôleur (RFx) transmettant des mesures au calculateur central du système (HUB). Chaque contrôleur utilise un routeur intégré sous forme de puce ce qui permet de garantir un débit constant à travers le réseau. Le réseau CAN est utilisé comme moyen de diagnostic et de programmation standard.

Le capteur développé par Autocruise est piloté par les RFs et les informations sont transférés directement au HUB qui les traitent. Dans une seconde itération du projet, les informations pourront aussi être pré-traités dans les RFs.

Contraintes posées

Chaque RF doit donc transférer l'ensemble des informations reçue du capteur sur le réseau. Après estimations des informations renvoyés par le capteur, le débit global nécessaire est de 44,8Mbps. L'application de démonstration devra donc atteindre ce débit.

L'ensemble de l'application doit être conforme aux spécifications AUTOSAR et devra servir d'exemple à l'apprentissage du personnel d'Autocruise.

Retours d'expérience

L'ensemble de mon stage m'a permis de mieux saisir les aspects cruciaux du développement d'un système embarqué et temps-réel. Les problèmes rencontrés tout au long de mon stage m'ont permis de trouver des solutions applicables dans d'autres cas. Je vous livre ici les points qui m'ont bloqué durant mon stage et leurs résolutions.

Carte EVK1100 : Atmel AVR32 famille UC3A

Cette carte d'évaluation m'a été fournie dès la 3ème semaine de stage. Dans le support de développement, la pile TCP/IP lwIP était fournie intégrée au système d'exploitation FreeRTOS. J'ai donc été confronté à la dépendance de la pile TCP/IP vis-à-vis des services d'un système d'exploitation.

En effet une pile TCP/IP utilise des mécanismes de transmissions et protections des messages qui permettent d'obtenir de bonnes performances. Cependant ces mécanismes sont généralement associés aux services du système d'exploitation et nécessite une adaptation par système.

Le système d'exploitation utilisé dans l'architecture AUTOSAR est OSEK¹³. Ce dernier n'est, comme AUTOSAR, que configurable de façon statique et n'offre pas de service de haut-niveau comme la gestion de messages dynamique. J'ai donc choisi dans un premier temps d'implémenter la pile sans système d'exploitation sur cette cible.

Après une étude approfondie des mécanismes nécessaires aux fonctionnalités j'en ai déduit un graphe de fonctionnalités. Au niveau TCP/IP, la pile ne nécessite qu'une gestion de l'allocation mémoire et de la fragmentation mémoire. Une implémentation d'une boîte aux lettres a cependant été nécessaire car l'interface de programmation POSIX en a besoin.

¹³ OSEK est un système d'exploitation développé par les constructeurs français et allemand. Il est la référence pour les ECU automobiles. AUTOSAR utilise OSEK pour tous les services qui concernent le système d'exploitation.

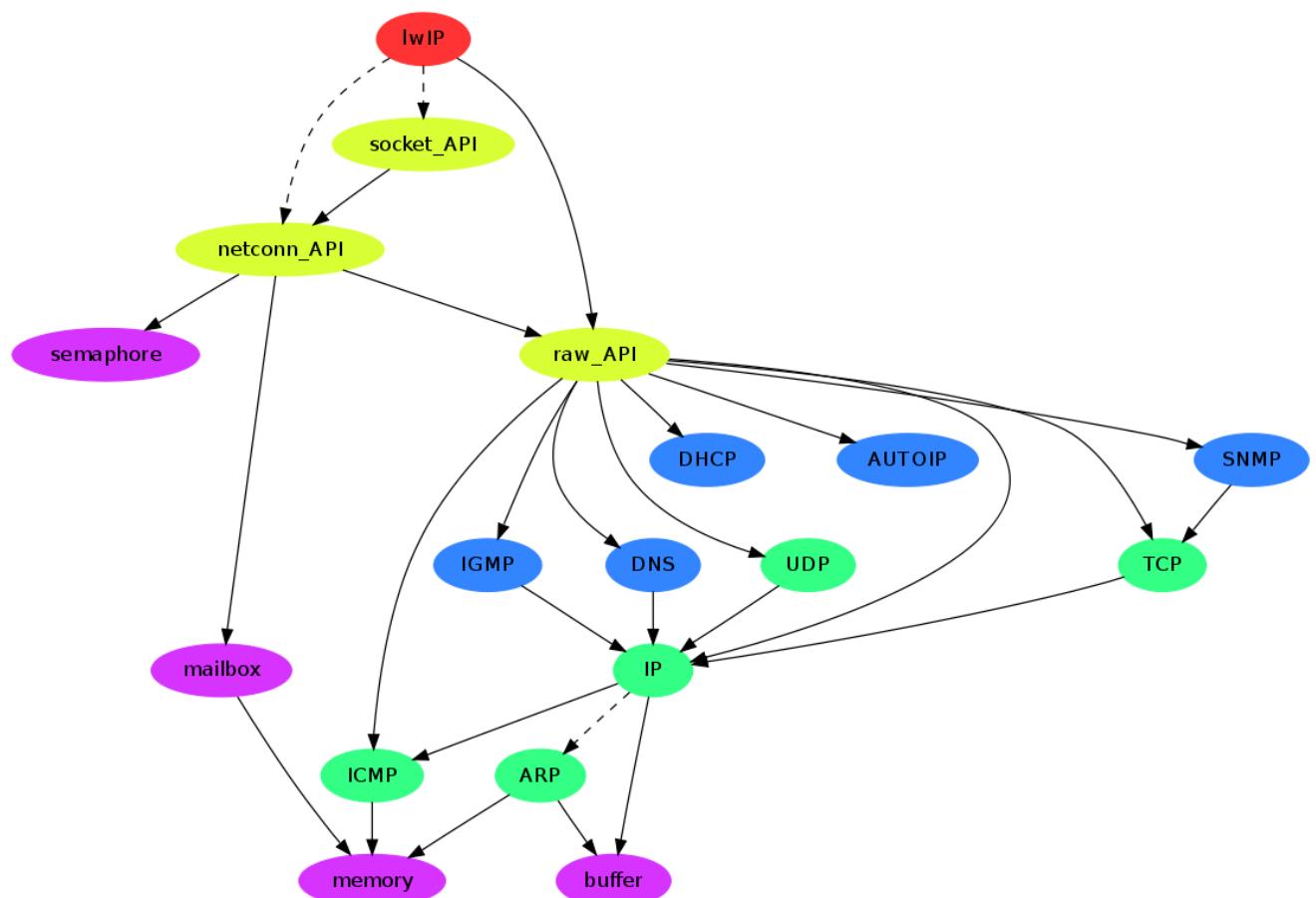


Illustration 24: Graphe de dépendance de la pile lwIP

Sur ce graphique, on observe que les dépendances matérielles sont : soit communes à n'importe quel système, soit nécessaires à l'utilisation des interfaces de programmation. Ces dernières ont donc été re-développées afin de répondre aux contraintes de l'architecture AUTOSAR. Les modules non spécifiés ont aussi été supprimés de l'implémentation.

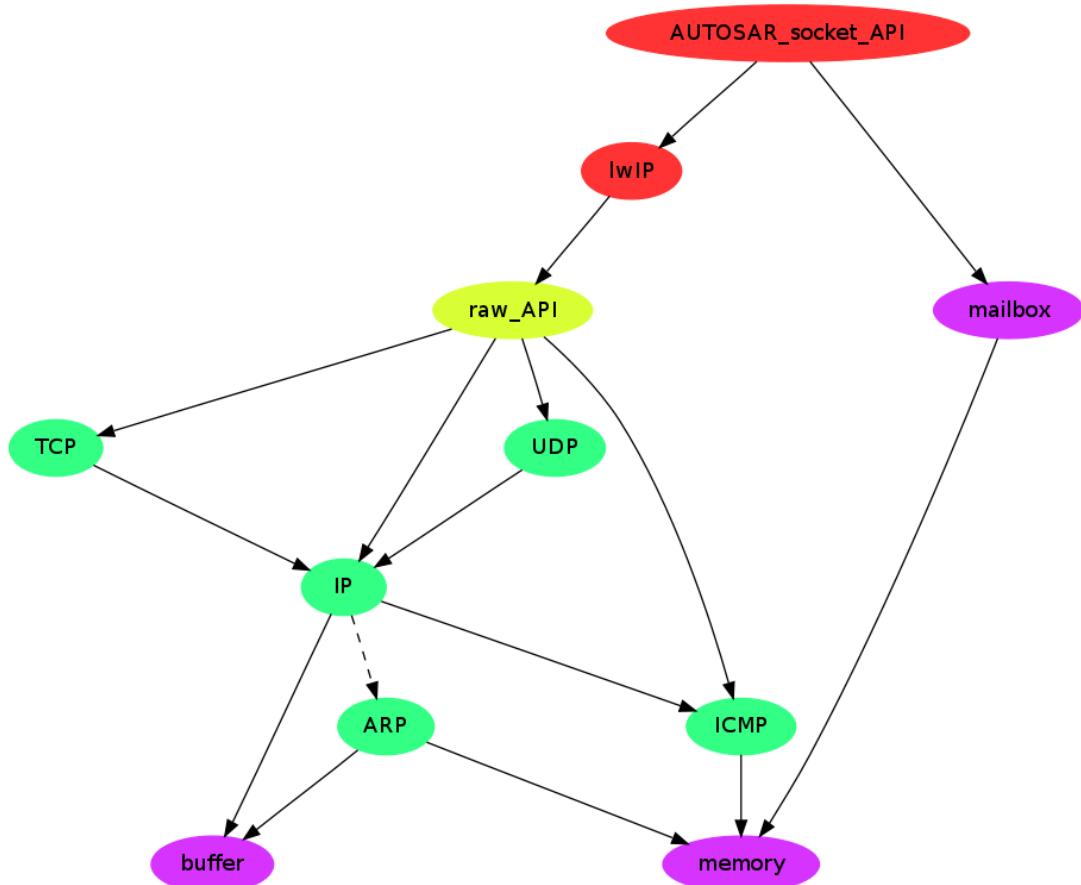


Illustration 25: Graphe de dépendance pour la solution AUTOSAR - lwIP

Afin de réaliser les premières intégrations, j'ai aussi utilisé la chaîne de compilation fournie par Atmel. L'outil AVR32 Studio étant instable et inutilisable dans la pratique, j'ai opté pour l'utilisation de l'outil Eclipse et de la chaîne de compilation en ligne de commande. Pour cela je me suis servi du système de construction 'GNU make' qui m'a permis d'effectuer les différentes phases d'intégration sans problèmes.

Développement du module suivant la méthode MCSE-Geensys

Le développement de module pour AUTOSAR suit la méthode MCSE. J'ai tout d'abord représenté la pile TCP/IP sous un certain formalisme. Ainsi, l'ensemble de l'interface de programmation POSIX 1003.1 a été décrite (voir illustration page 30). La méthode MCSE offre une vue de haut niveau où les moyens de communication sont typés. Ce sont : soit des variables partagées, soit des files d'attentes, soit des événements.

Afin de suivre ce formalisme j'ai choisi de représenter l'ensemble de l'interface à travers des variables partagées. En effet, un descripteur de fichier¹⁴ peut être vu comme une variable partagée qui permet à deux modules logiciels de communiquer entre eux.

De plus, une exigence de qualité impose un taux de couverture de tests modulaires d'au moins 80%. Afin de garantir cette exigence, j'ai utilisé l'outil 'GNU gcov', le système de compilation 'GNU make' et le compilateur 'GNU Gcc' sur une cible débarquée. Les différents cas et séquences de tests ont nécessité une construction différente du module (configuration par pré-compilation) ce qui pose un problème à l'outil 'GNU gcov'. En effet celui-ci ne permet d'extraire des informations résumées que lors d'un seul cas d'utilisation uniquement.

Afin de résoudre ce problème j'ai développé un script pour 'GNU awk' qui joint l'ensemble des fichiers de sortie de l'outil 'GNU gcov' et en affiche un résumé. J'ai ainsi pu valider l'exigence du taux couverture. Le script est joint en annexe de ce document et a été proposé au service qualité de la société.

¹⁴ Un descripteur de fichier est, d'après la norme POSIX, un identifiant que l'on crée à l'établissement de la communication, qui sera détruit à la fin de la communication. C'est l'instance du lien de communication.

Carte MPC5668 : Freescale PowerPC famille MPC55xx/MPC56xx

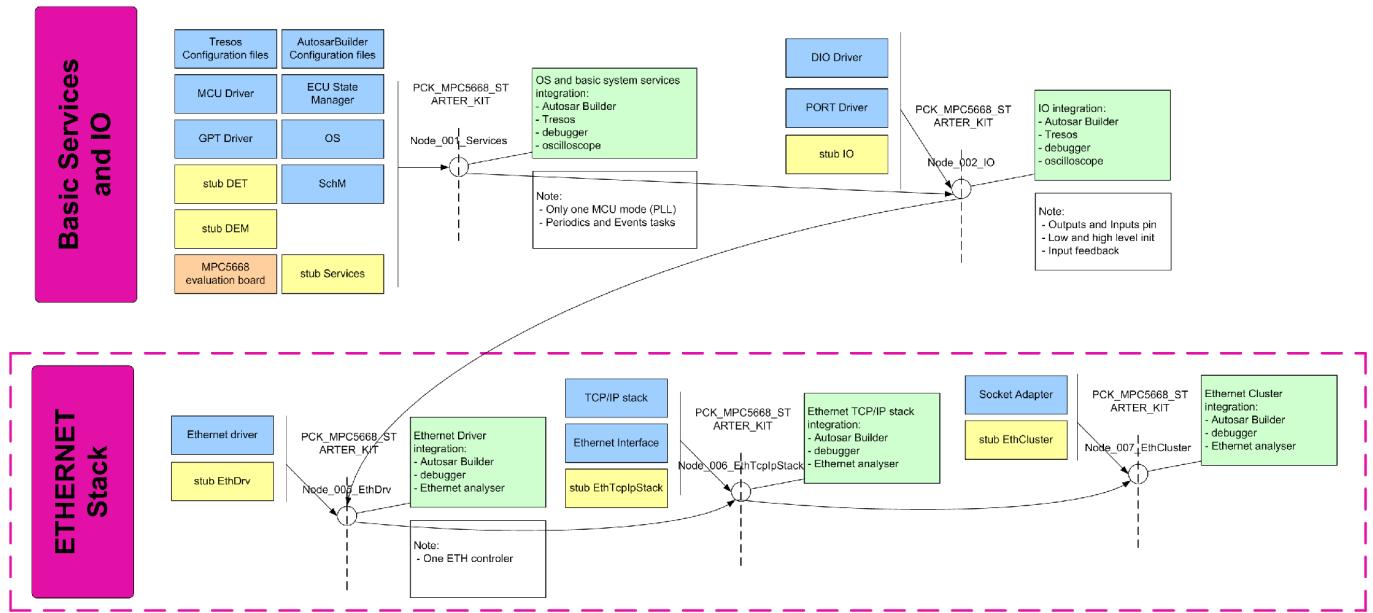


Illustration 26: Schéma d'intégration de la pile TCP/IP pour MPC5668

Tout d'abord, sur cette nouvelle cible, j'ai développé le pilote du contrôleur Ethernet. La spécification de la cible est, à l'heure actuelle, toujours sous forme d'ébauche. J'ai donc bloqué sur certains points qui étaient non cohérents avec la carte de démonstration fournie. Par exemple, certains registres du contrôleur Ethernet étaient marqué comme initialisés au démarrage alors qu'en réalité ceux-ci n'était pas initialisés.

De plus le schéma de la carte de démonstration ne correspondait pas au câblage. J'ai donc dû la résoudre les problèmes matériels afin que celle-ci puisse transmettre des informations à la vitesse souhaitée.

Différents problèmes d'intégration ont ensuite été résolus. On peut notamment citer des problèmes de configuration du compilateur, d'alignements d'accès mémoires et de configuration des allocations.

L'intégration a aussi été ardue à cause des limitations de l'outil de programmation et de débogage. En effet dans le cadre du projet ARPOD, Autocruise devait livrer à Geensys un émulateur performant. Cependant celui-ci n'a pas été livré et l'ensemble de l'intégration a été rendue difficile par ce manque.

Lors de mon départ de Geensys, l'intégration était rendu au noeud 007 qui consiste à intégrer le module SoAd, la pile TCP/IP et le driver dans l'architecture AUTOSAR. Un retard de deux semaines est ainsi intervenu suite aux problèmes rencontrés.

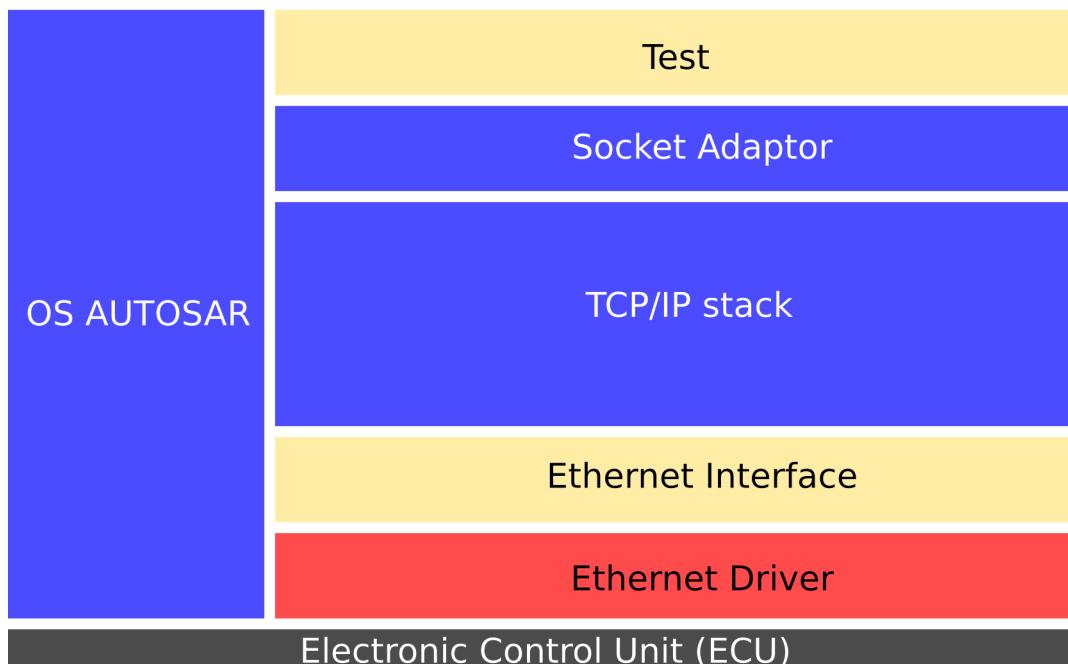


Illustration 27: Diagramme de l'architecture à l'intégration du noeud 007

Conclusion

Ces 6 mois passés à Geensys m'ont apporté énormément tant sur le plan technique que sur le plan personnel. En effet j'ai pu m'intégrer à l'ensemble de l'équipe AUTOSAR et ainsi, dialoguer avec des personnes d'expériences.

Geensys, exemple d'une équipe soudée est une société qui m'a assuré une bonne intégration. L'organisation des activités permet de garantir au mieux le déroulement des développements. D'ailleurs, l'ensemble de cette société est structuré de façon à suivre les personnes tout en leur laissant un périmètre d'actions libres.

Par conséquent, cette autonomie de travail m'a aidé à mieux approfondir les méthodes et outils métiers. La méthode MCSE et le processus de développement CMMI-3 ont pu être étudiés et appliqués de façon réfléchie. Une vision du système a aussi été nécessaire notamment lors du développement de la couche d'interface POSIX.

De plus, j'ai pu découvrir lors de mon stage l'évolution des métiers du domaine des systèmes embarqués. À travers les réunions hebdomadaires, j'ai eu un point de vue global des actions en cours. L'intérêt au travail des autres membres de l'équipe est aussi amélioré grâce aux réflexions techniques issues du partage d'expériences.

Pour conclure, durant cette période, j'ai apprécié les fonctions d'encadrement, l'environnement technologique et surtout l'ambiance d'équipe qui permettent de travailler dans d'excellentes conditions et qui favorisent le succès des projets.

Glossaire

- 5 Electronic Control Unit : Système ou Sous-Système électronique généralement basé sur un micro-contrôleur.
- 5 Une pile TCP/IP est un composant logiciel implémentant les couches de communications du modèles OSI.
- 24 Commercial Off The Shelf : Solution sur étagère
- 24 Protocol Data Unit Router : Réalise la liaison entre la couche de communication applicative et les différents réseaux
- 25 Ethernet Network Manager : voir p.27 : Architecture choisie
- 25 Ethernet State Manager : voir p.27 : Architecture choisie
- 26 Domain Name Service : Protocole qui permet d'associer un nom à chaque composant du réseau.
- 26 Internet Group Management Protocol : Permet de réaliser du multicast sur réseau IP.
- 26 Simple Network Management Protocol : Outils d'administration des noeuds du réseau, permet de diagnostiquer et configurer les équipements du réseau.
- 28 L'algorithme de Nagle permet de diminuer la bande passante occupée par les entêtes TCP en retardant l'émission de données utiles.
- 30 La méthode MCSE (Méthodologie de Conception des Systèmes Électroniques) est une méthode de conception conjointe développée par Jean-Paul Calvez à l'Université Polytechnique de Nantes.
- 32 Protocol Data Unit : Message contenant des données et leur taille.
- 37 OSEK est un système d'exploitation développé par les constructeurs français et allemand. Il est la référence pour les ECU automobiles. AUTOSAR utilise OSEK pour tous les services qui concernent le système d'exploitation.
- 40 Un descripteur de fichier est, d'après la norme POSIX, un identifiant que l'on crée à l'établissement de la communication, qui sera détruit à la fin de la communication. C'est l'instance du lien de communication.

Bibliographie

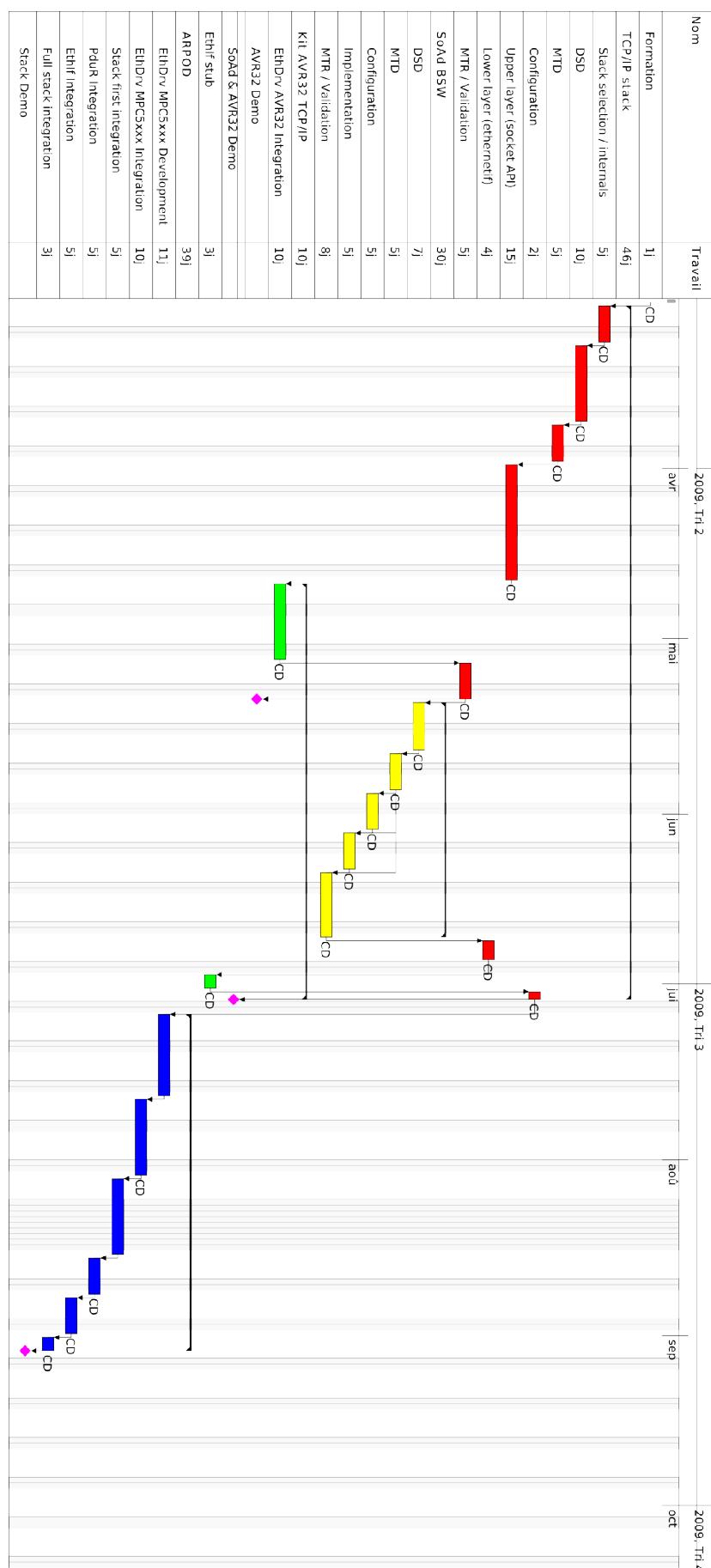
- Site de la société Geensys, consulté le 03/08/2009, <http://www.geensys.com>
- Site du consortium AUTOSAR, consulté le 03/08/2009, <http://www.autosar.org>
- Site pôle ID4CAR, consulté le 03/08/2009, <http://www.id4car.org>
- Site de l'IETF (Internet Engineering Task Force), consulté le 03/08/2009, <http://www.ietf.org>
- Site de l'IEEE (Institute of Electrical and Electronics Engineers), consulté le 03/08/2009, <http://standards.ieee.org/>
- Standard de l'IEEE dédié aux réseaux locaux (accessible librement), consulté le 03/08/2009, <http://standards.ieee.org/getieee802/portfolio.html>
- Documentation de la carte de démonstration ATMEL pour AVR32, consulté le 03/08/2009, http://www.atmel.com/dyn/Products/tools_card.asp?tool_id=4114
- Site central de développement de la pile lwIP, consulté le 03/08/2009, <http://savannah.nongnu.org/projects/lwip/>
- Wiki de développement de la pile lwIP, consulté le 03/08/2009, <http://lwip.wikia.com>
- Site d'Adam Dunkels, auteur original de la pile lwIP, consulté le 03/08/2009, <http://www.sics.se/~adam/lwip/>
- Site du projet GNU, développant les outils 'GNU make', 'GNU Gcc', 'GNU gcov', 'GNU awk', consulté le 03/08/2009, <http://www.gnu.org>

Annexes

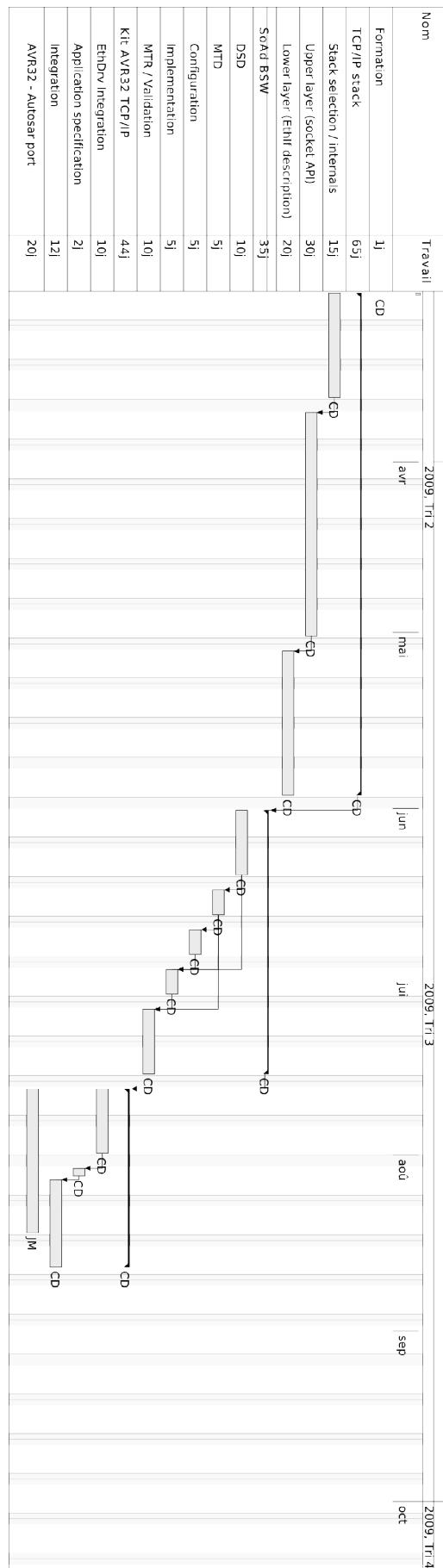
Annexe 1: Planning réel.....	2
Annexe 2: Planning prévu.....	3
Annexe 3: Script de jonction de plusieurs fichiers gcov.....	4
Annexe 4: Diagramme de séquence en réception.....	5
Annexe 5: Diagramme de séquence en transmission.....	6
Annexe 6: Photo de la carte d'évaluation EVK1100 pour AVR32.....	7
Annexe 7: Photo de la carte d'évaluation pour MPC5668 (Freescale).....	8

Document joint

- Support de présentation



Annexe 1: Planning réel



Annexe 2: Planning prévu

```

#!/usr/bin/gawk -f
#
# Parse *.gcov files passed as arguments and print result file on stdout and
# summary on stderr.
#
# View summary on windows: ./merge-gcov.awk SoAd.c.gcov >NUL
# Concatenate files: ./merge-gcov.awk my-dir1/SoAd.c.gcov my-dir2/SoAd.c.gcov >SoAd.c.gcov
#
BEGIN {
    FS=":"
}
#
# Non passed line
#
$1 ~ /#####\'/ {
    sum_call[$2] += 0;

    if (ARGIND == ARGC-1)
    {
        if (sum_call[$2] != 0)
            printf "%5d:%5d:%s\n", sum_call[$2], $2, $3;
        else
            print;
    }
}

#
# Comment line
#
$1 ~ /-\\'/ {
    if (ARGIND == ARGC-1)
        print;
}

#
# Summary line
#
$1 ~ /function (\w)+ called [0-9]+ returned [0-9]+% blocks executed [0-9]+%\'/ {
}

#
# Passed line
#
$1 ~ /[0-9]+\\'/ {
    sum_call[$2] += $1;

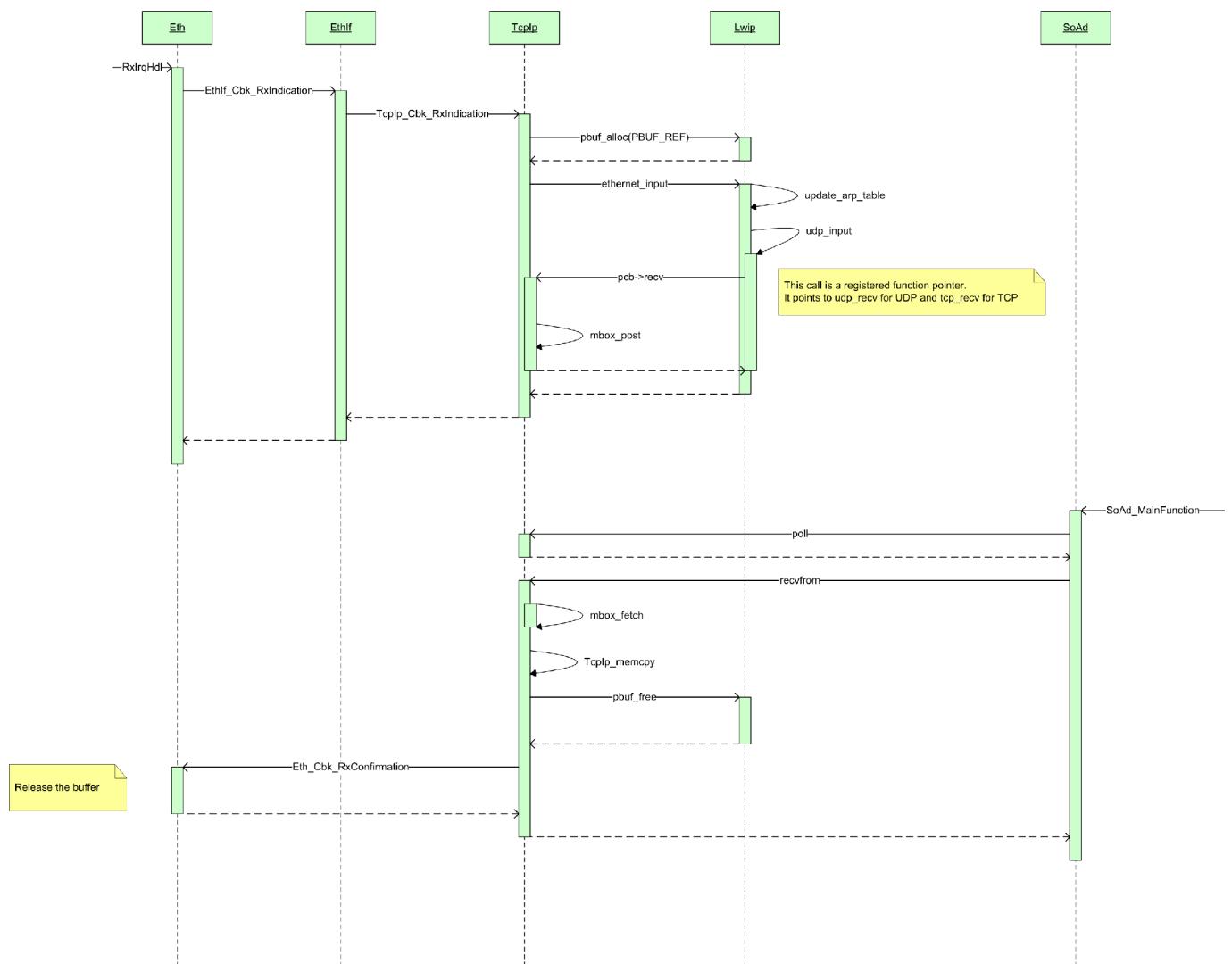
    if (ARGIND == ARGC-1)
    {
        if (sum_call[$2] != 0)
            printf "%5d:%5d:%s\n", sum_call[$2], $2, $3;
    }
}

END {
    all_lines = length(sum_call);
    for(i in sum_call)
    {
        if (sum_call[i] > 0)
            passed_lines++;
        else
            non_passed_lines++;
    }

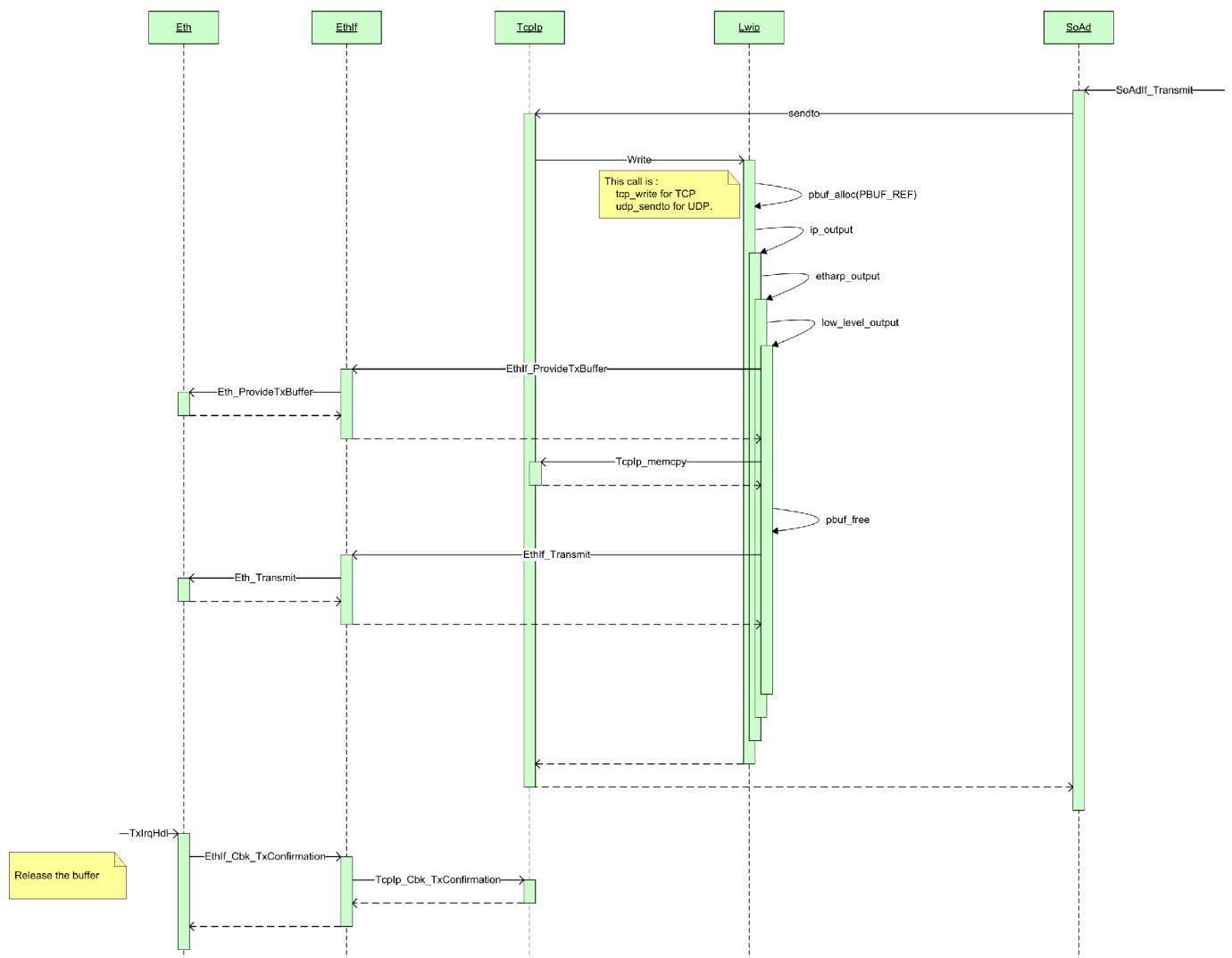
    print "Line executed: " ((passed_lines*100)/all_lines) "% of " all_lines >/dev/stderr";
}

```

Annexe 3: Script de jonction de plusieurs fichiers gcov



Annexe 4: Diagramme de séquence en réception



Annexe 5: Diagramme de séquence en transmission



Annexe 6: Photo de la carte d'évaluation EVK1100 pour AVR32



Annexe 7: Photo de la carte d'évaluation pour MPC5668 (Freescale)

Clément DAVID
91, rue de Brissac
49000 Angers

Port : **06 26 26 51 90**
Tél : 02 99 49 04 61
e-mail : clement.david@reseau.eseo.fr

Né le 10 Mai 1986
Permis B
Voiture personnelle

Objectif : Stage de fin d'étude

Études :

- 2004 – 2009 : École d'ingénieur en électronique et informatique. ESEO (Angers) option Systèmes Embarqués et Automatismes (SEA).
- 2001 – 2004 : Baccalauréat S option : Science de l'ingénieur (équivalent bac E)

Expériences Scolaires :

- 2008 Dans le cadre d'un projet de fin d'étude, réalisation d'un banc de test CAN logiciel sur plateforme Eclipse [<http://www.eclipse.org>].
- 2008 Stagiaire Hardware pour VALEO Lighting System à Angers (49). Conception d'une carte de diagnostic pour éclairage à LEDs à l'aide de Spice pour le monde automobile.
- 2007 Développeur unique d'un module pour un site internet de qualité industriel dans le cadre d'une junior-entreprise en C# (.NET).
- 2003 – 2005 Travail durant les vacances comme Ouvrier de Production

Expériences Open source :

- Utilisateur Linux depuis 2003, compétence multi-distributions : LFS, RedHat, Debian, Ubuntu, ...
- Langages de programmation et concepts maîtrisés : C, Python, Java, VHDL, C#, .NET, intérêt pour la validation formelle
- Projets
 - Vala : Langage haut niveau type C# générant du code C en utilisant les conventions objets de Gnome.
Responsable de [<http://code.google.com/p/vala-benchmarks>]
 - Gnome, ... : Environnement de bureau pour systèmes POSIX
Rapport de bug, patch et utilisation des composants.

Divers :

Vie Associative à l'ESEO

- Secrétaire du V.V.V. (Association de découverte Internationale)
- Trésorier puis Président d'un CinéClub.

Anglais (TOEIC : 765 et FIRST : C)
Basket-Ball (pratiqué depuis l'âge de 8 ans)

Annexe 8: CV utilisé pour postuler à ce stage



geensys

Global Embedded Electronics & Networked System Solutions

Intégration d'une pile TCP/IP comme composant AUTOSAR

Clément David - Stage de fin d'étude 2009



© Geensys – Clément David

1



Sommaire



- **La société Geensys**
- **La consortium AUTOSAR**
- **Le sujet de stage**
- **Les résultats du stage**
- **Questions / Réponses**

- **La société Geensys**
 - Métiers
 - Solutions
 - Chiffres
- **La consortium AUTOSAR**
- **Le sujet de stage**
- **Les résultats du stage**
- **Questions / Réponses**

Intégration d'un pile TCP/IP dans AUTOSAR

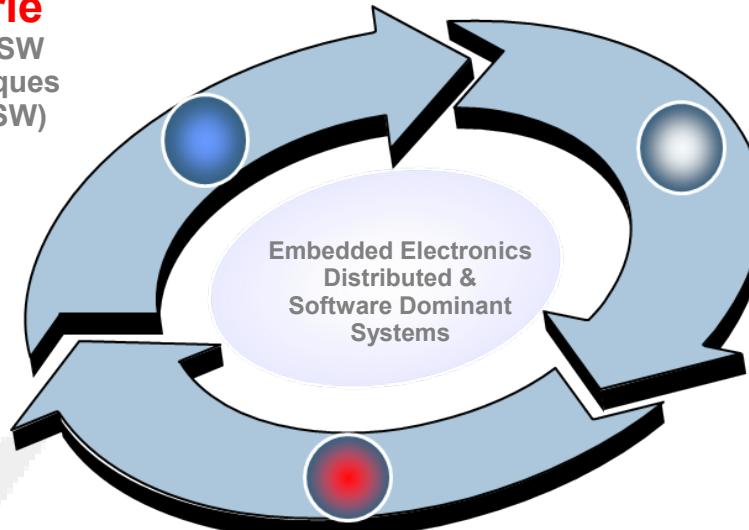
3

4 métiers complémentaires : l'outillage pour le développement, l'électronique, l'informatique industrielle et la mécanique.

- **Hardware** : Numérique, analogique, puissance.
- **Software** : Pour systèmes embarqués et débarqués.
- **Mécanique** : Développement, design et industrialisation de moules et pièces plastiques, prototypage rapide.
- **Mécatronique** : Associe des organes mécaniques et électroniques, alliance de nos savoir-faire projets pour une Vision Système.

Ingénierie

Projets HW, SW
et mécatroniques
Blocs d'IP (SW)

**Outils et chaînes outillées**

Solutions
centrées exigences
et modèles

Services et Conseil

Projets et Process

Intégration d'un pile TCP/IP dans AUTOSAR

5

- **Création : 2001**
- **6 sites : Carquefou (siège), Brest, Boulogne et Nancy**
- **CA fin 2008 : 20 M€**
- **Clients : 60 clients actifs**
- **Croissance : 30% par an 3 années consécutives (2006-2007-2008)**
- **Effectif total : 249 personnes (97% d'ingénieurs)**

- **La société Geensys**
- **La consortium AUTOSAR**
 - Le consortium
 - Acteurs
 - Architecture
- **Le sujet de stage**
- **Les résultats du stage**
- **Questions / Réponses**

Intégration d'un pile TCP/IP dans AUTOSAR

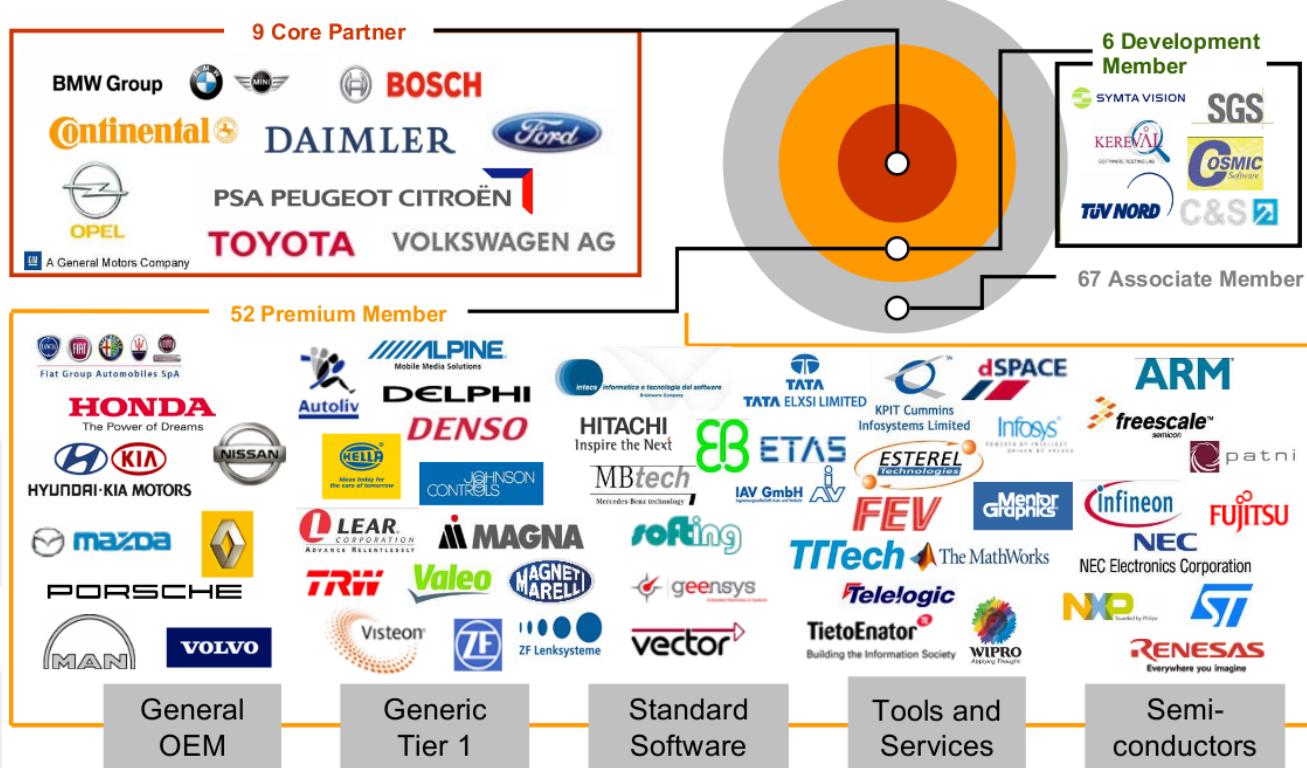
7

AUTOSAR – Le consortium

- **AUTomotive Open System Architecture**
- Consortium international
- **Objectif** : Standardisation des architectures des calculateurs électroniques.
- **Composition du consortium :**
 - Constructeurs automobiles
 - Équipementiers
 - Bureaux d'études en développement logiciel
 - Fabricant de semi-conducteurs

AUTOSAR – Acteurs

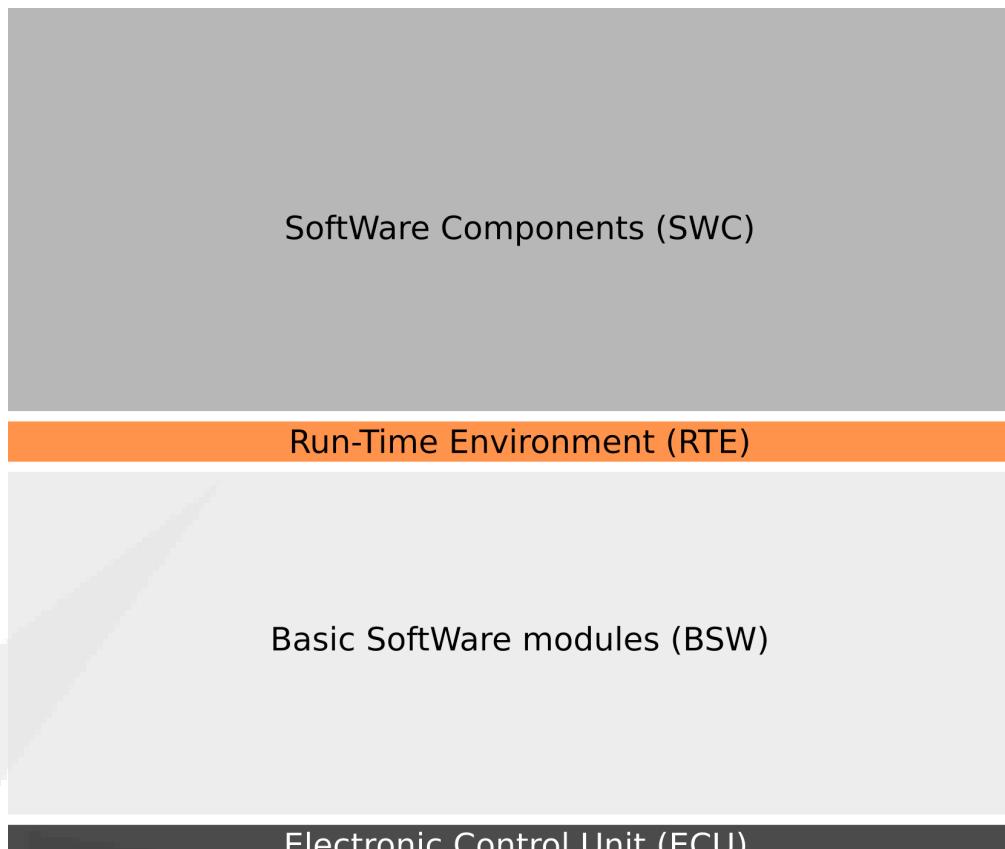
Status: 13th March 2008



Intégration d'un pile TCP/IP dans AUTOSAR

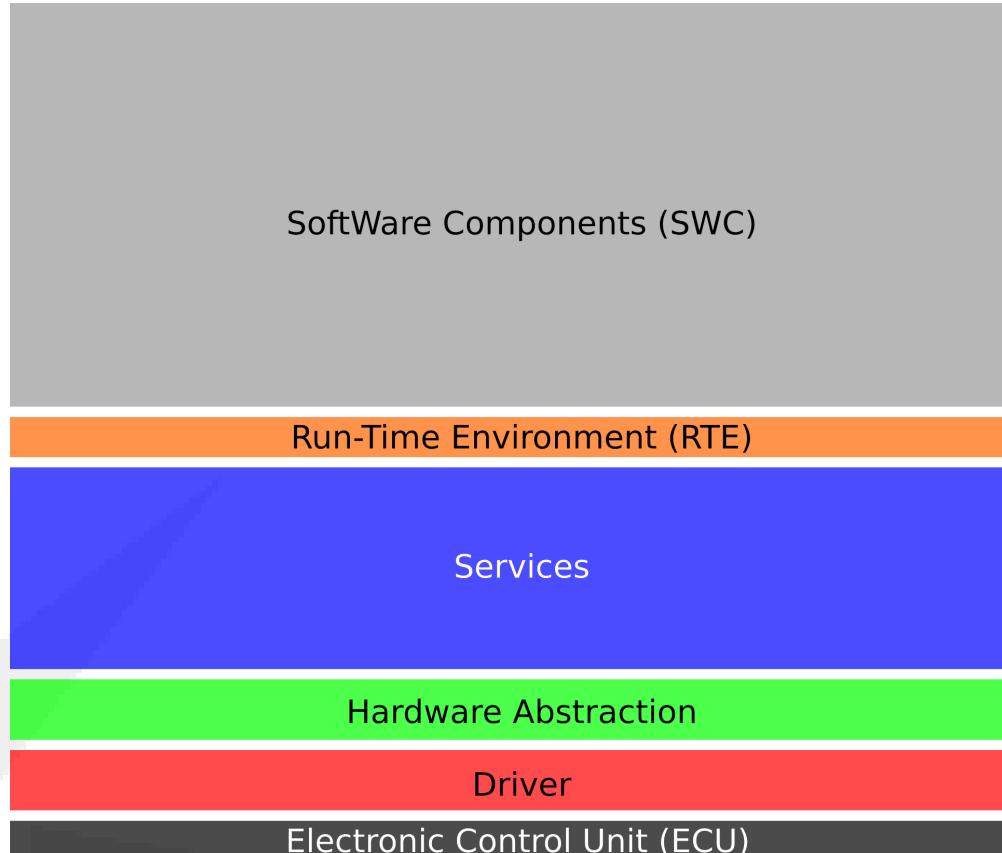
9

AUTOSAR – Architecture



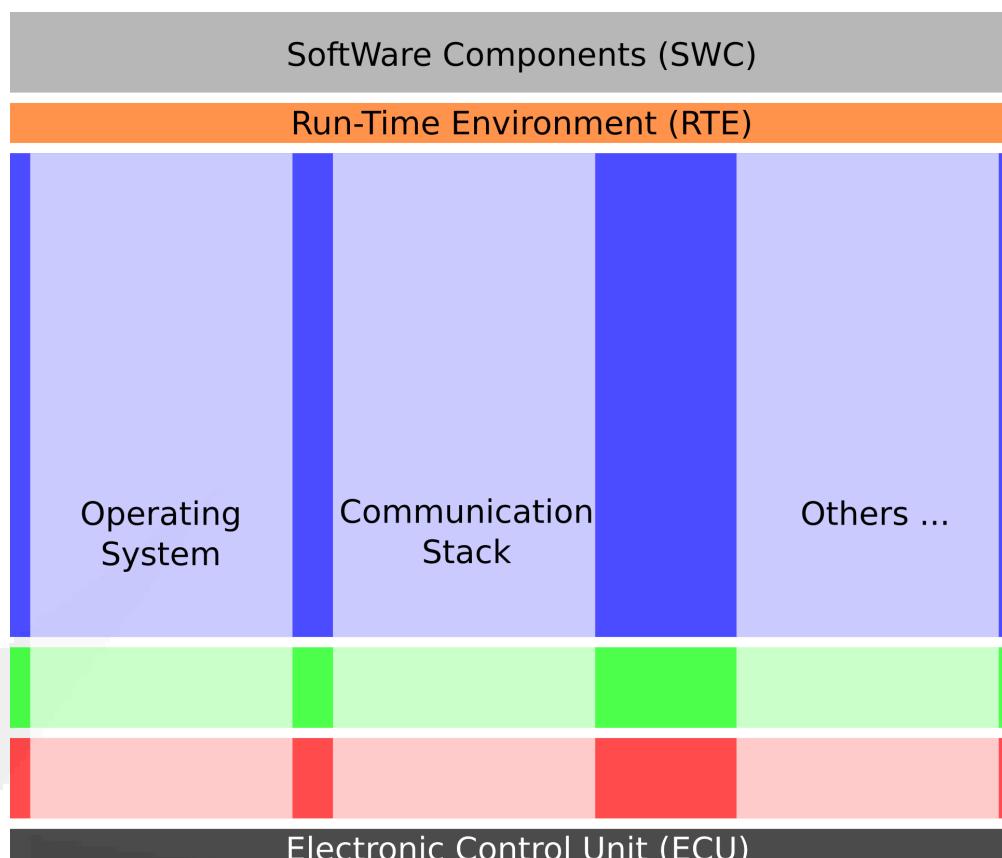
Intégration d'un pile TCP/IP dans AUTOSAR

10



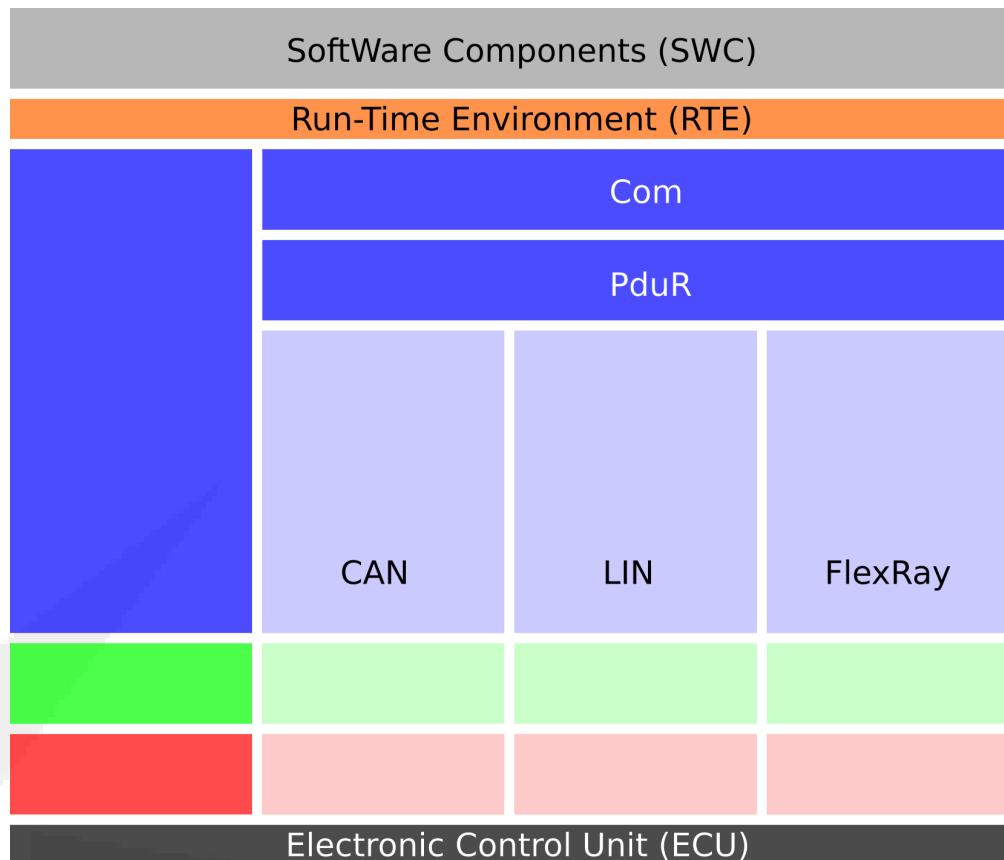
Intégration d'un pile TCP/IP dans AUTOSAR

11



Intégration d'un pile TCP/IP dans AUTOSAR

12



Intégration d'un pile TCP/IP dans AUTOSAR

13

Sujet de stage

- **La société Geensys**
- **La consortium AUTOSAR**
- **Le sujet de stage**
 - Objectifs
 - Arpod
 - Vue schématique
 - Architecture globale
 - Planning effectif
- **Les résultats du stage**
- **Questions / Réponses**

Sujet de stage – Objectifs

- **Objectifs initiaux :**

- Sélectionner une pile TCP/IP Open-source répondant aux exigences AUTOSAR.
- Développer un ensemble de BSW permettant de supporter du TCP/IP sur Ethernet.
- Réaliser une application de test sur un démonstrateur.

Intégration d'un pile TCP/IP dans AUTOSAR

15

Sujet de stage – Objectifs

- **Objectifs initiaux :**

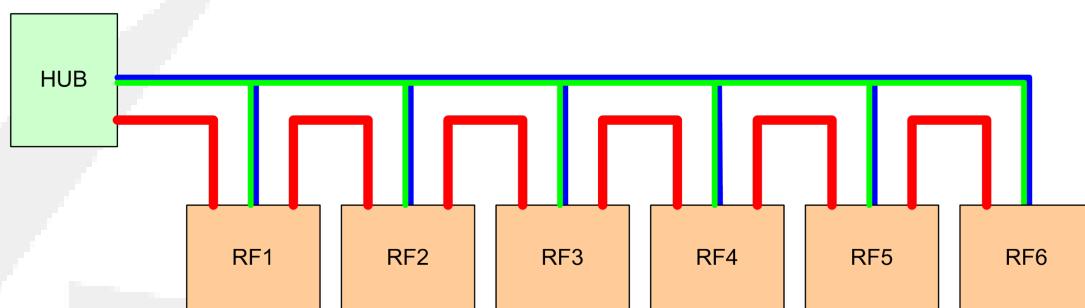
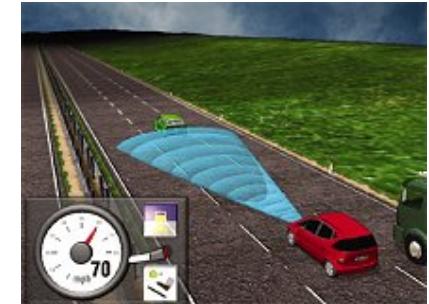
- Sélectionner une pile TCP/IP Open-source répondant aux exigences AUTOSAR.
- Développer un ensemble de BSW permettant de supporter du TCP/IP sur Ethernet.
- Réaliser une application de test sur un démonstrateur.

- **Objectifs réalisés :**

- Sélectionner une pile TCP/IP Open-source répondant aux exigences AUTOSAR.
- Développer un ensemble de BSW permettant de supporter du TCP/IP sur Ethernet.
- **Développer les couches basses de l'Ethernet pour cible MPC5668**
- **Intégrer l'ensemble des modules pour le projet ARPOD.**

« Développer un pare choc intelligent, intégrant des capteurs miniaturisés, associés à une unité centrale de traitement.

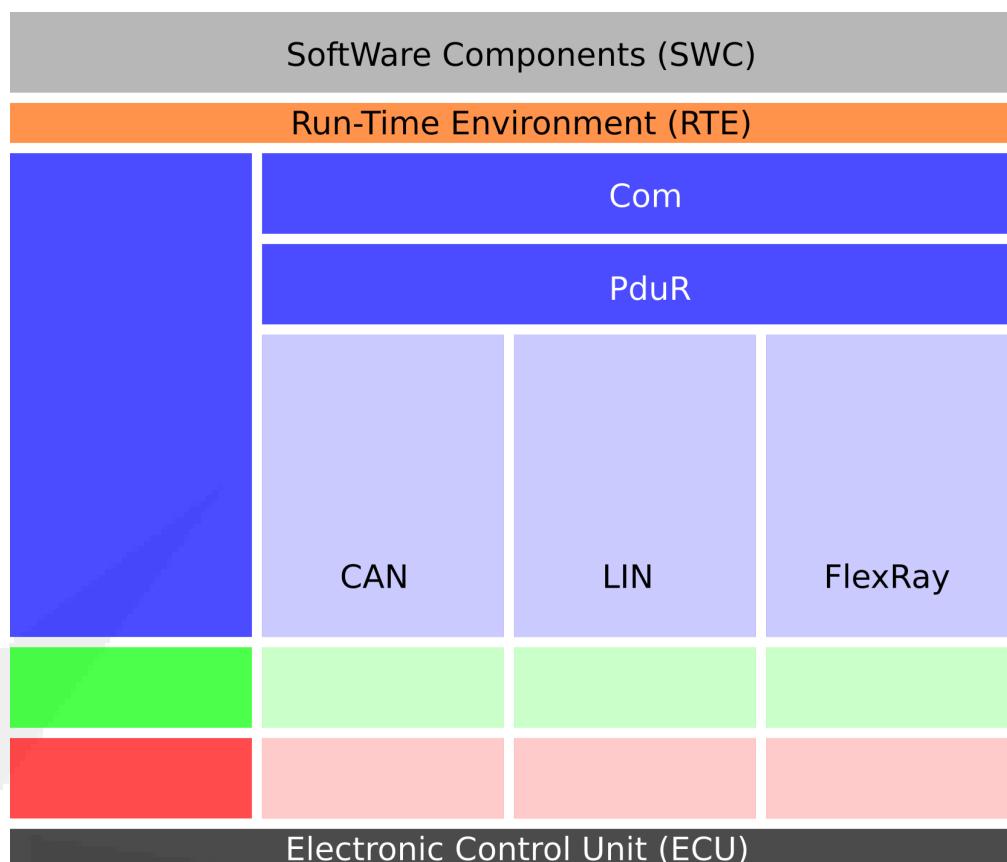
L'objectif est d'améliorer la perception périphérique de l'environnement du véhicule pour un meilleur contrôle du véhicule. »



Intégration d'un pile TCP/IP dans AUTOSAR

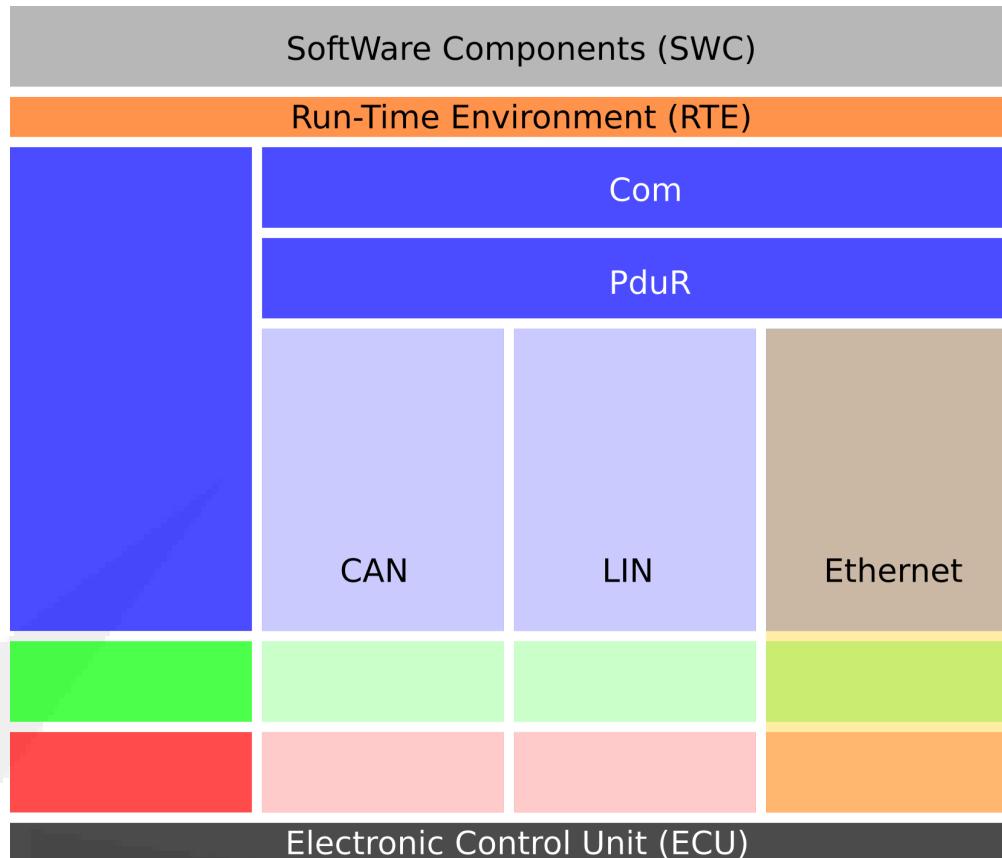
17

Projet – Vue schématique



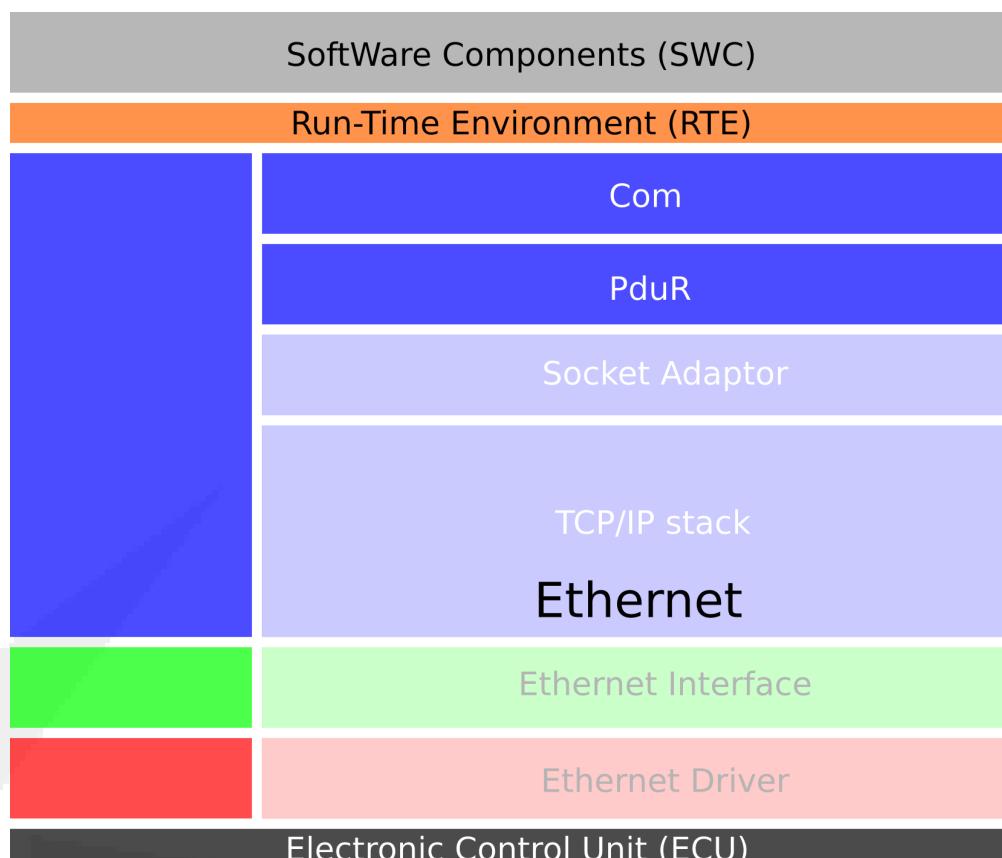
Intégration d'un pile TCP/IP dans AUTOSAR

18



Intégration d'un pile TCP/IP dans AUTOSAR

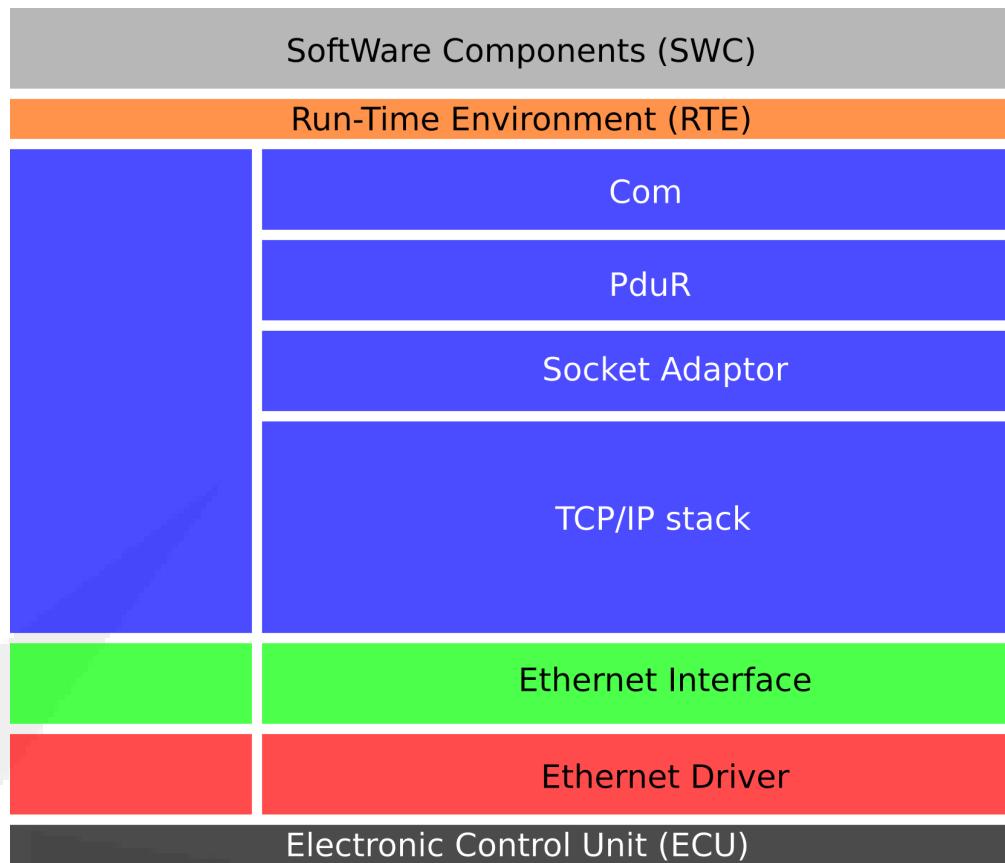
19



Intégration d'un pile TCP/IP dans AUTOSAR

20

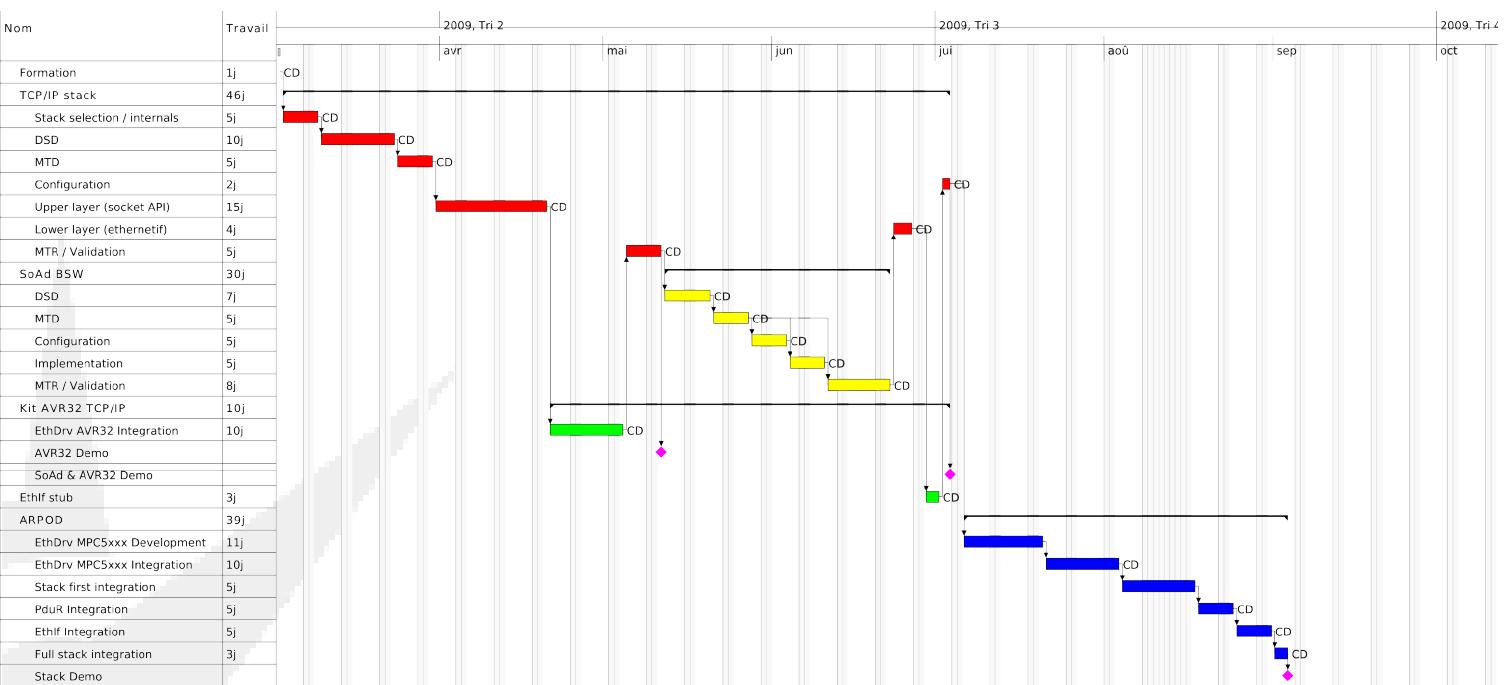
Projet – Architecture globale



Intégration d'un pile TCP/IP dans AUTOSAR

21

Projet – Planning effectif



Intégration d'un pile TCP/IP dans AUTOSAR

22

- **La société Geensys**
- **La consortium AUTOSAR**
- **Le sujet de stage**
- **Les résultats du stage**
 - Carte ATMEL AVR32
 - Module AUTOSAR : SoAd
 - Carte Freescale MPC5668
 - Conclusion
- **Questions / Réponses**

Intégration d'un pile TCP/IP dans AUTOSAR

23

- **Sélection d'une pile TCP/IP open-source**
 - Configurable à la compilation
 - Compatible avec le standard IEEE 1003.1
 - License non contraignante
- Pour cibles embarquées
- Portable sur les plateformes 16 ou 32 bits

uIP (micro IP)	IwIP (lightweight IP)	*BSD (FreeBSD)
Configuration statique	Configuration statique	Configuration dynamique
Spécifique	Standard IEEE 1003.1 (partiel) et spécifique	Standard IEEE 1003.1 (complet)
BSD	BSD	BSD
Embarquable facilement	Embarquable facilement	Nécessite un travail important pour l'embarqué
Facilement portable	Facilement portable	Suppose des fonctionnalités du système
8 bits	16 ou 32 bits	32 bits

Intégration d'un pile TCP/IP dans AUTOSAR

25

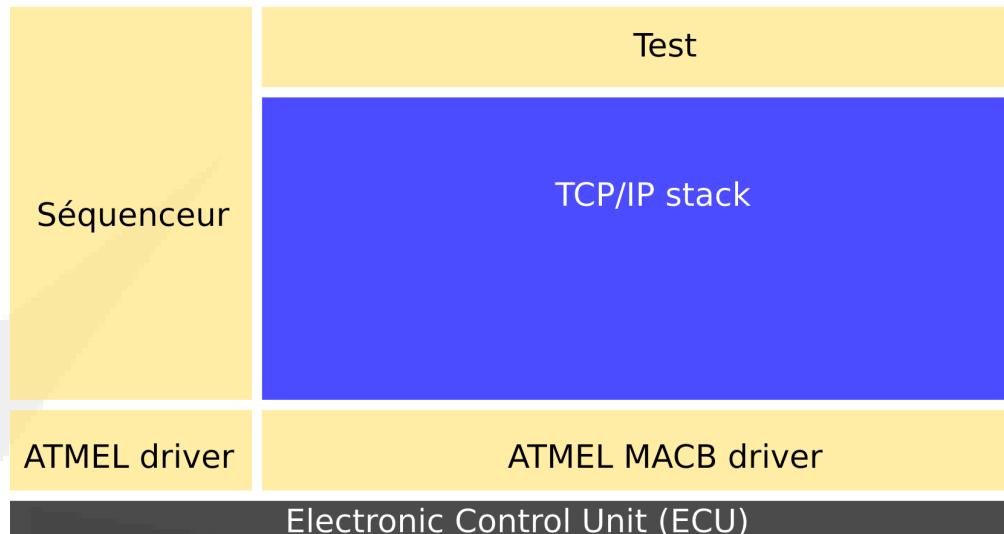
- Sélection d'une pile TCP/IP open-source
- Architecture simple
- Réutilisation du driver ATMELc



Intégration d'un pile TCP/IP dans AUTOSAR

26

- Sélection d'une pile TCP/IP open-source
- Architecture simple
- Réutilisation du driver ATMEL



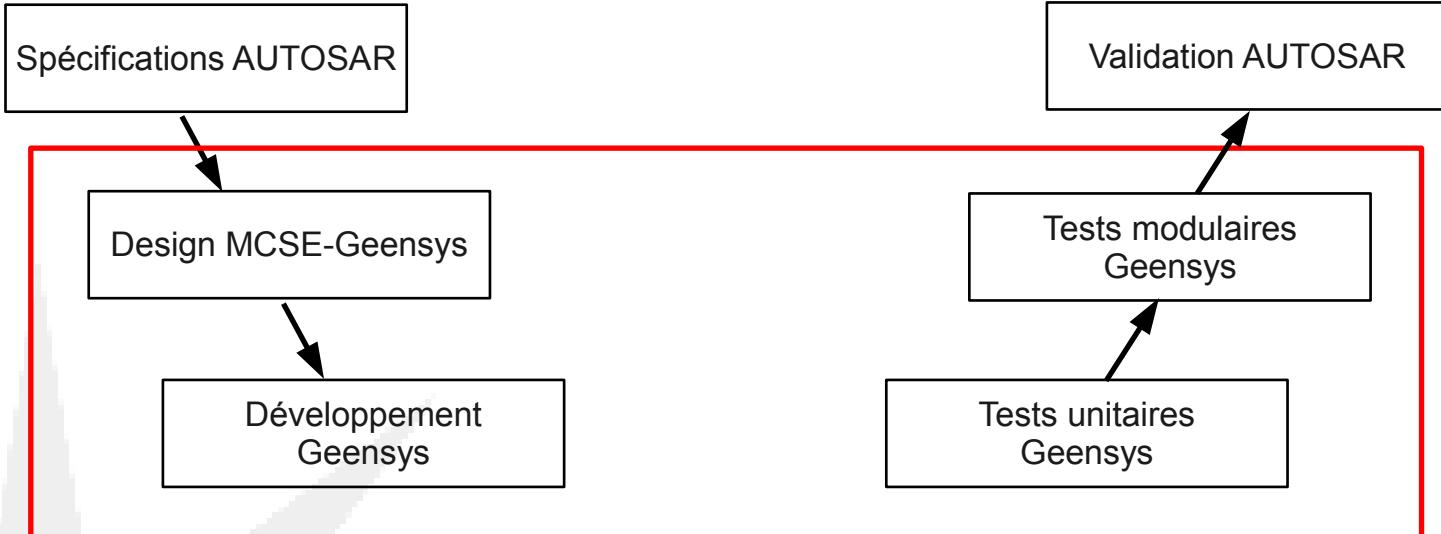
Intégration d'un pile TCP/IP dans AUTOSAR

27

Compétences – ATMEL AVR32

- GNU gcc sur AVR32
- Portage de lwIP
- Contraintes d'architecture AUTOSAR
- Notions réseaux étendues





Intégration d'un pile TCP/IP dans AUTOSAR

29

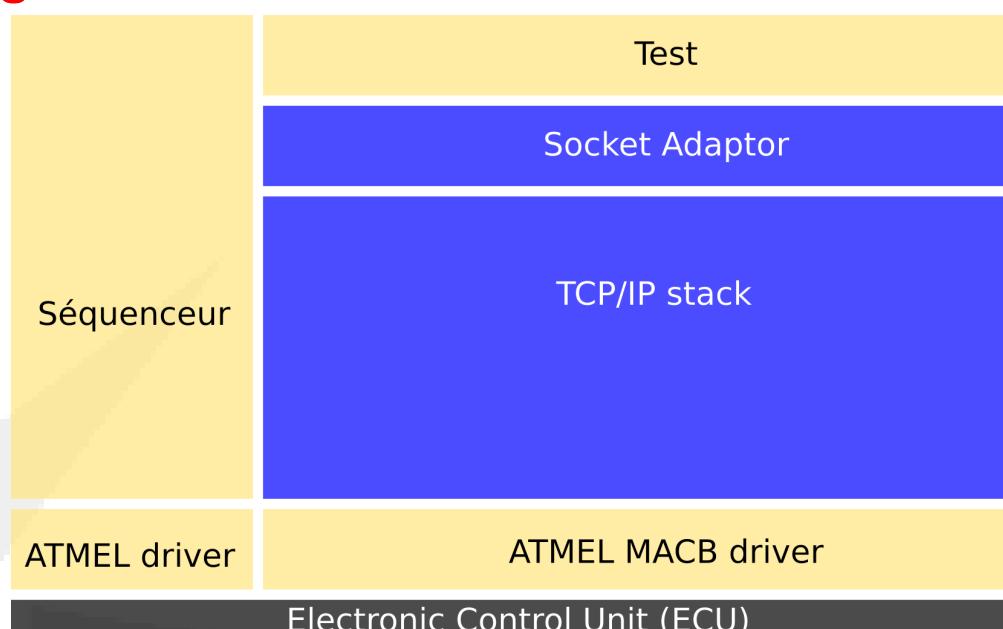
- **Documents de développement :**
 - DSD : Detailed Software Design
 - MTD : Modular Test Description
 - MTR : Modular Test Report
- **Couverture de test : 88,7%**
 - Outil de construction automatique
 - Outil de tests automatique
 - Outil de couverture de tests automatique
- **Développement sur ordinateur débarqué**

- MISRA-C
- Architecture AUTOSAR
- Autosar Builder
- GNU gcc, GNU make, GNU gcov, GNU awk
- IEEE 1003.1 (POSIX socket)
- Consortium AUTOSAR

Intégration d'un pile TCP/IP dans AUTOSAR

31

- **Architecture simple**
- **Intégration du module**



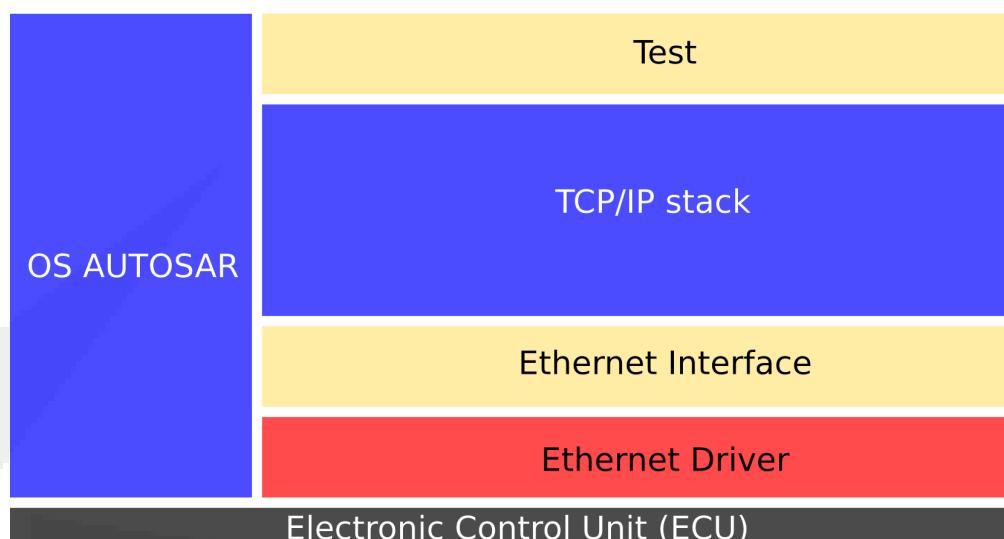
- Développement du driver Ethernet
- Architecture AUTOSAR



Intégration d'un pile TCP/IP dans AUTOSAR

33

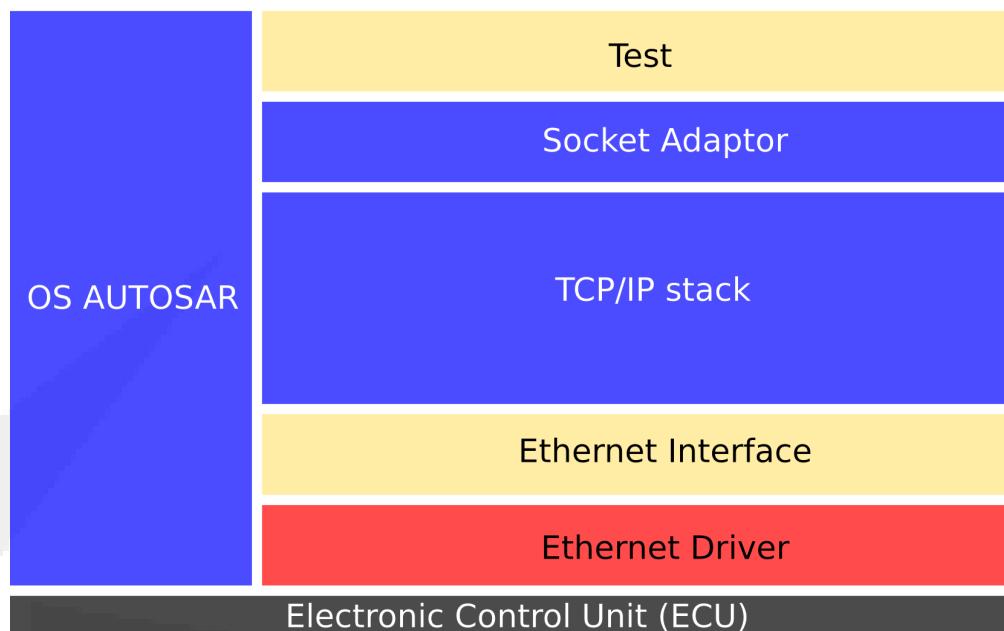
- Intégration de la pile TCP/IP
- Architecture AUTOSAR



Intégration d'un pile TCP/IP dans AUTOSAR

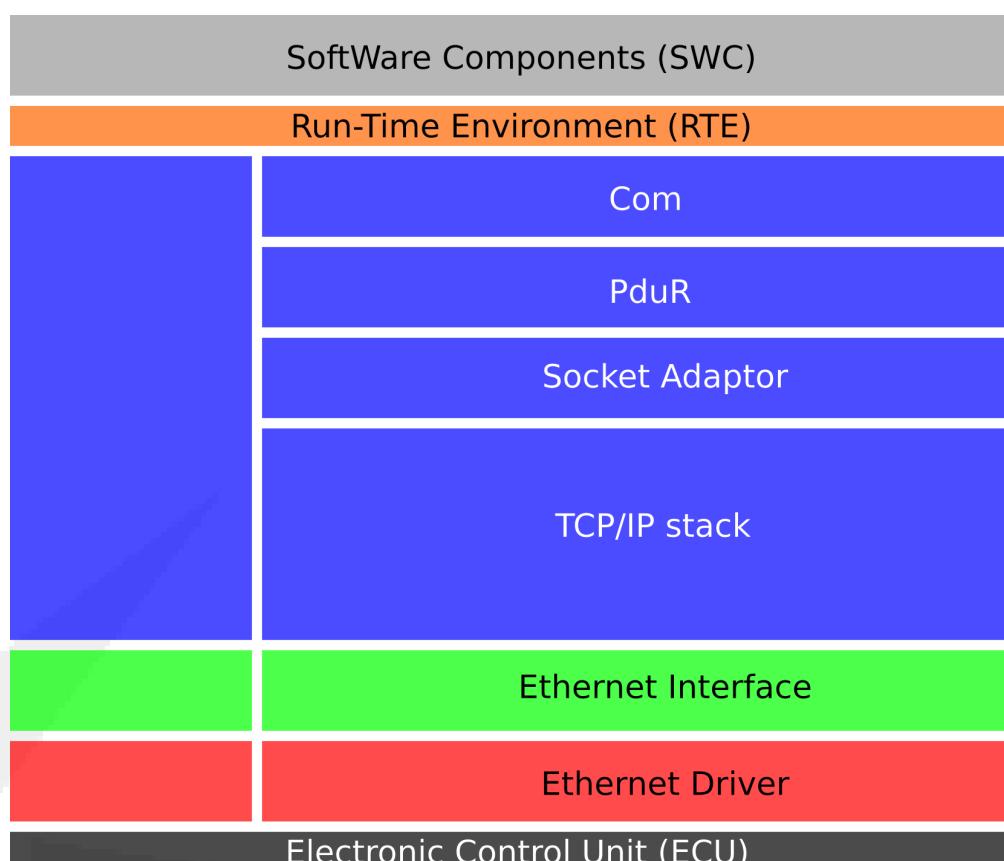
34

- **Intégration du module SoAd**
- **Architecture AUTOSAR**



Intégration d'un pile TCP/IP dans AUTOSAR

35



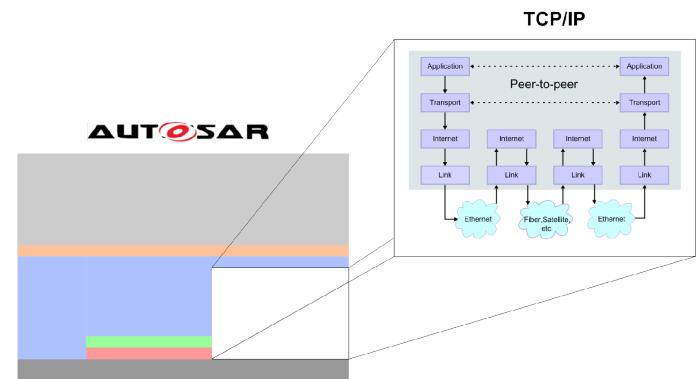
Intégration d'un pile TCP/IP dans AUTOSAR

36

Conclusion

Livrables

- Module SoAd
 - Module Tcplp (porté)
 - Module EthDrv
 - Adaptation PduR
-
- Script jonction gcov
 - Revue SoAd SWS

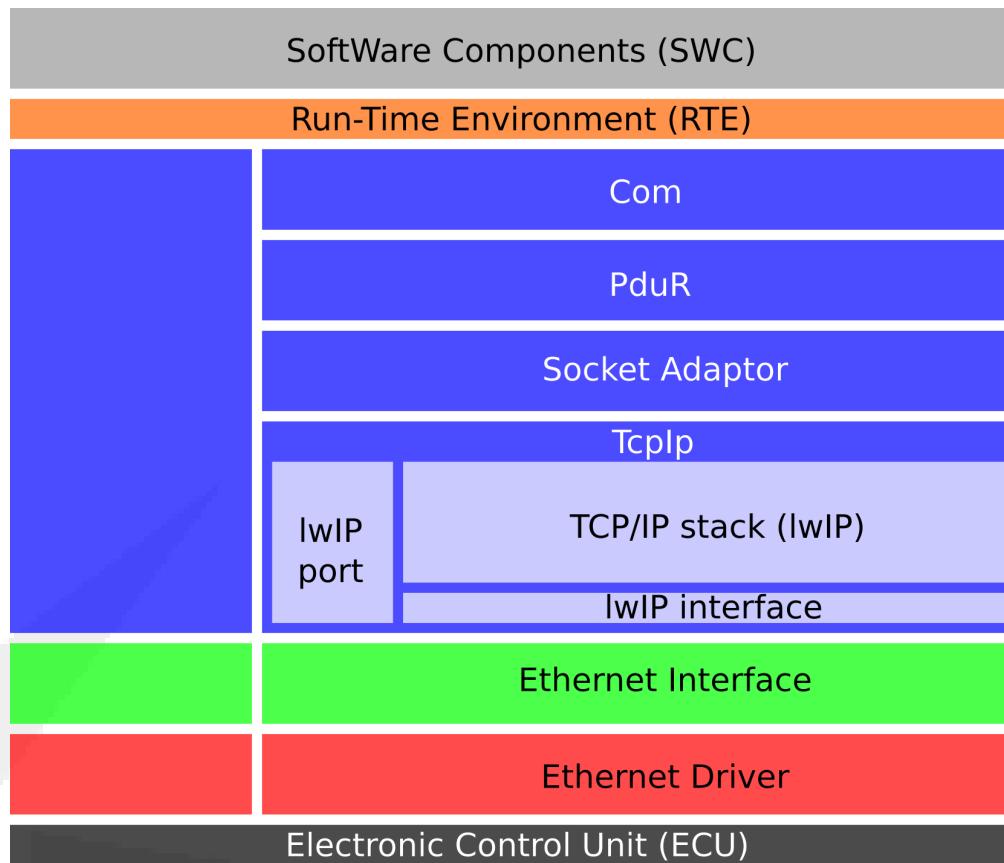


Intégration d'un pile TCP/IP dans AUTOSAR

37

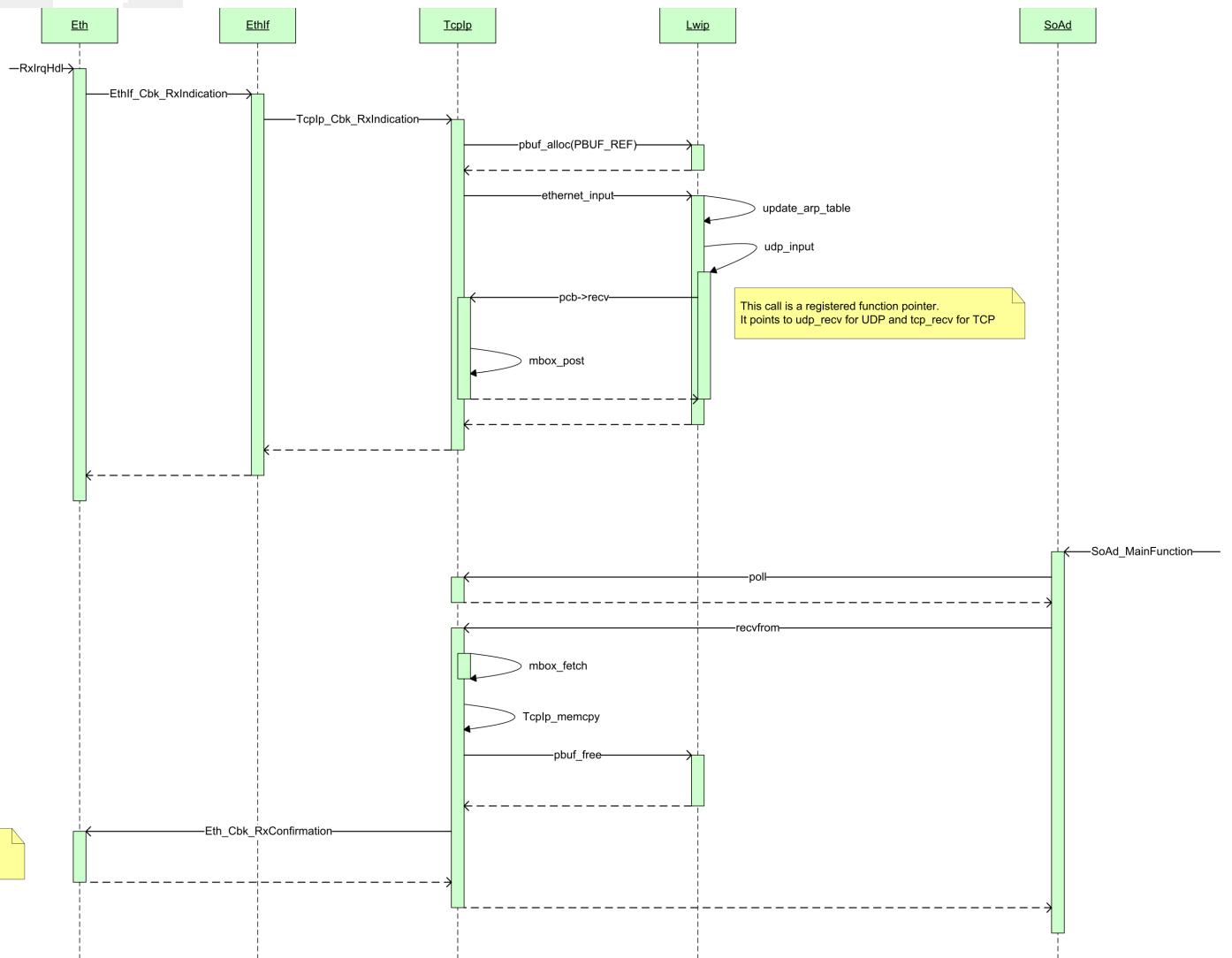
Merci de votre attention.

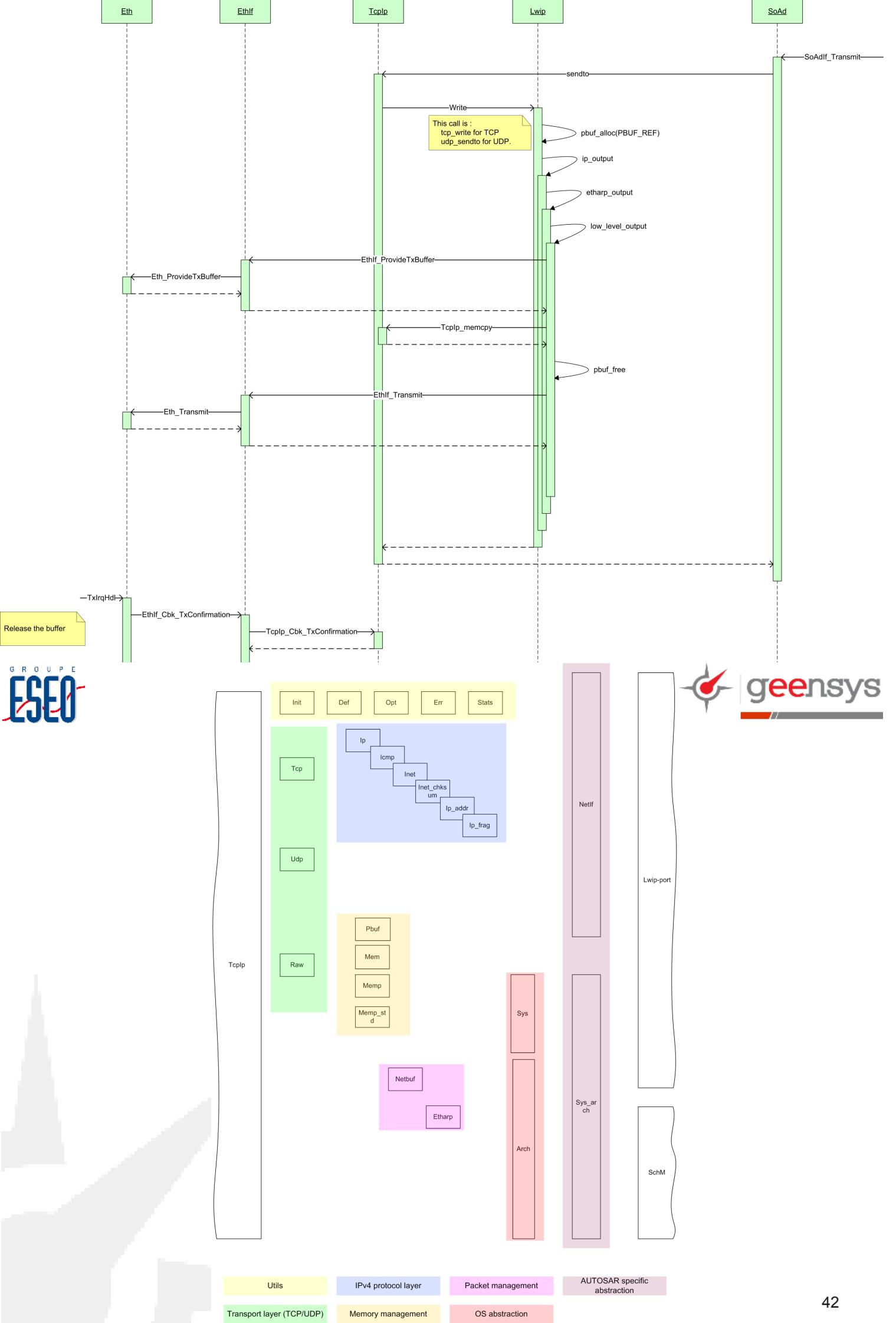
Questions ?

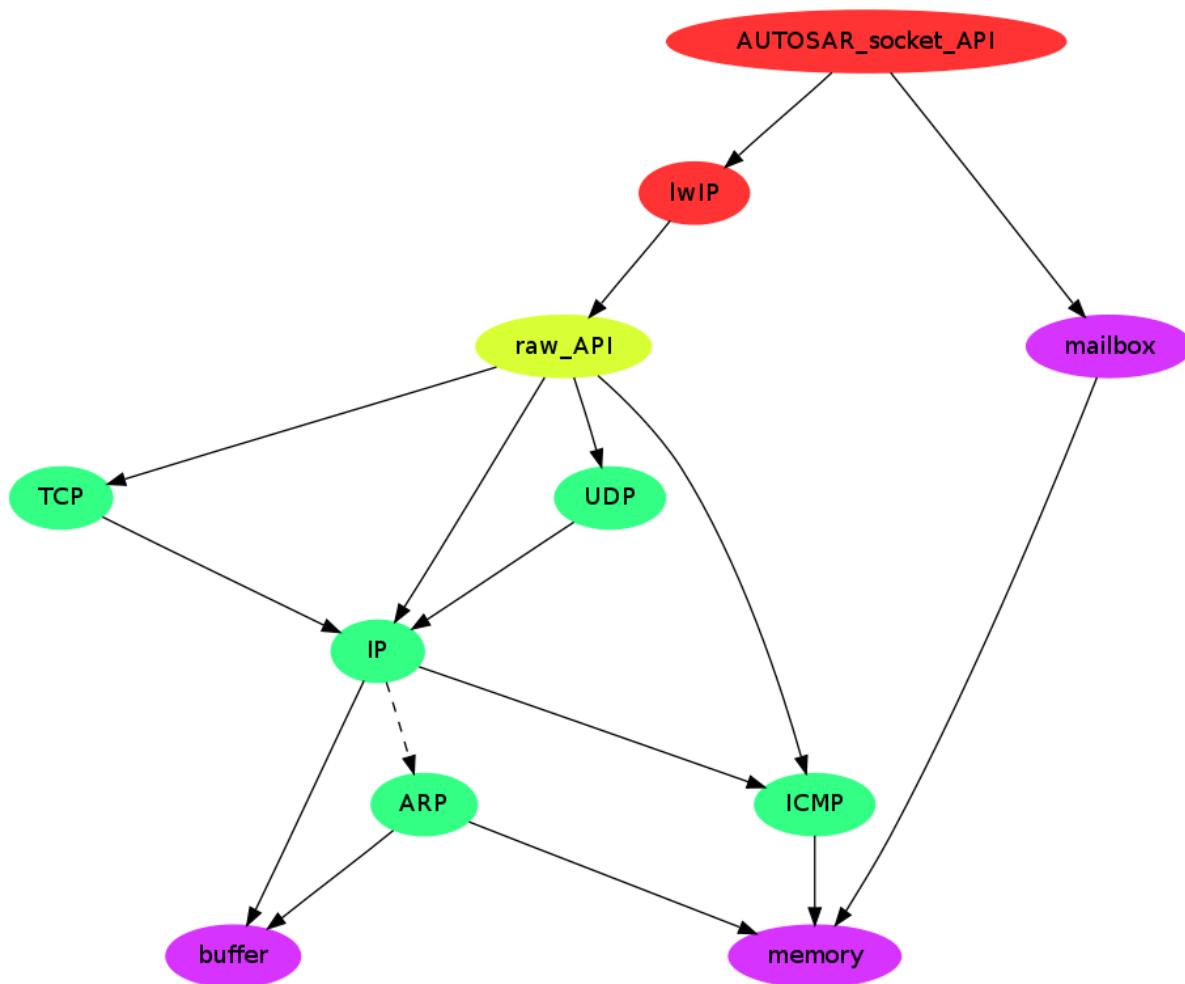
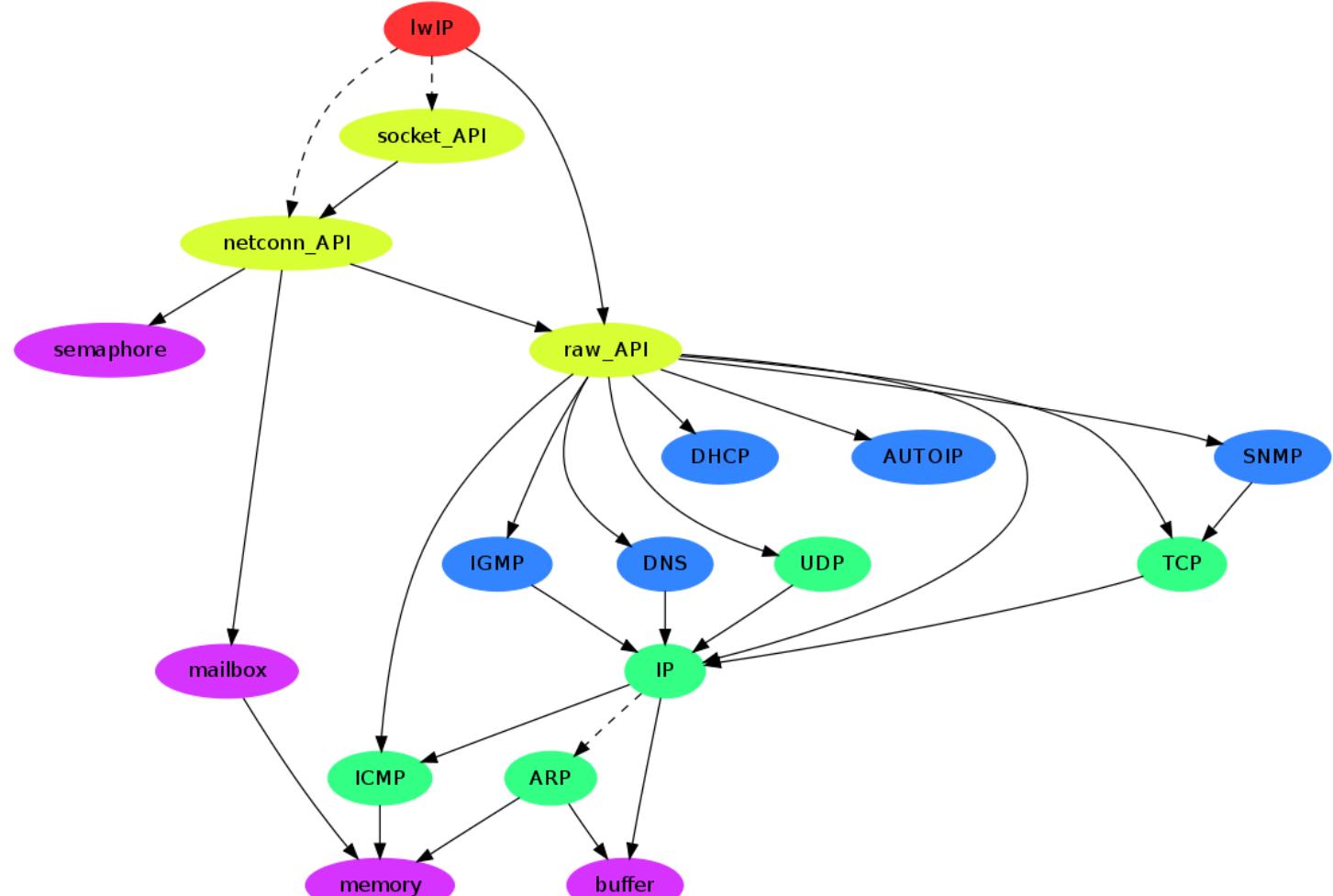


Intégration d'un pile TCP/IP dans AUTOSAR

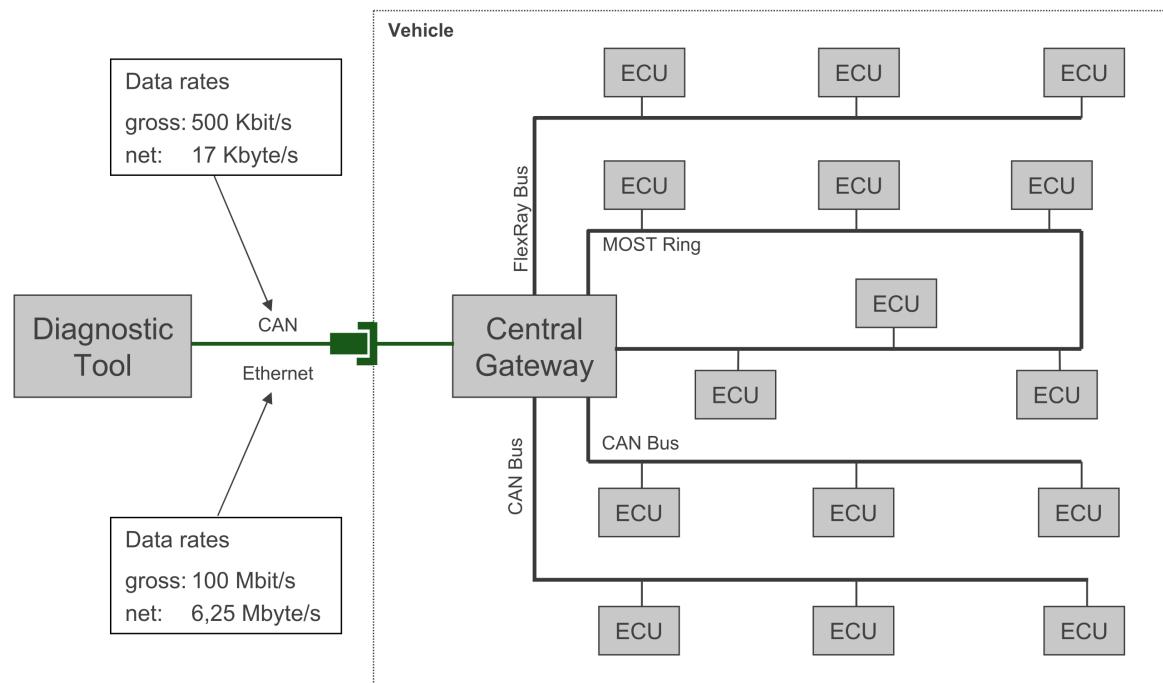
39







Typical Network Topology



Copyright Elektrobit Corporation 2008
Elektrobit.com, 23-Oct-08, Slide 2

