



Ouest Plateforme Automobile Systèmes embarquéS

Réseaux de terrain CAN, LIN et FLEXRAY



Ouest Plateforme Automobile Systèmes embarquéS

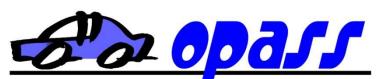
- Présentation de la plateforme OPASS
- Présentation générale
- Présentation du bus Can
- Présentation des protocoles J1939 / FMS, ISOBUS, NMEA2000
- Présentation du protocole CANopen
- Présentation du bus Lin
- Présentation du bus Flexray



Ouest Plateforme Automobile Systèmes embarquéS

Présentation Plateforme OPASS

3



Ouest Plateforme Automobile Systèmes embarquéS

Avantages de la plate-forme

- Matériel mutualisé

- Matériel performant et complet (bus CAN, LIN, Flex Ray, Most)
- Temps réel
- Configuration à jour et matériel prêt à l'emploi
- Formations spécifiques
- Compétence technique
- paiement à la prestation





Clients potentiels

- Entreprises TPE, PME, PMI, SSII
- Equipementiers, constructeurs automobiles
- Ecoles professionnelles / universités



2 axes principaux

- Modélisation
- Communication (Can, Lin, FlexRay, Most)

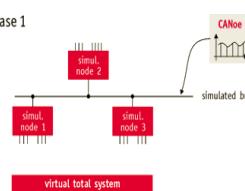
opass
Ouest Plateforme Automobile Systèmes embarqués

CANoe (Vector)

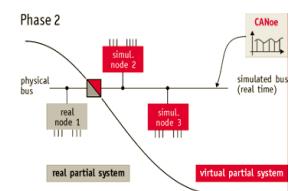
Simulation et analyse de réseaux
Can (ISOBUS, J1939, CANopen, NMEA2000), Lin, Most et FlexRay



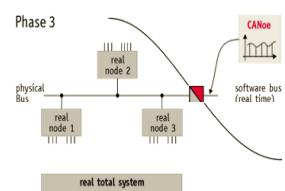
Phase 1



Phase 2



Phase 3



opass
Ouest Plateforme Automobile Systèmes embarqués

Oscilloscope (LeCroy)

Analyse des trames transitant sur un bus Can, Lin ou FlexRay



INTRODUCTION GÉNÉRALE



9

Introduction Générale (1/3)

- BUT INITIAL :

=> Remplacement des boucles *analogiques* de courant 4 - 20 mA

MAIS AUSSI :

=> Distribution (décentralisation) du contrôle, du traitement des alarmes, diagnostics au différents équipements de terrain

=> Intelligence déportée au niveau de ces équipements

=> Interopérabilité (système ouvert ?)



10

Introduction Générale (2/3)

Besoins de l'Automobile:

- **Commande d'actionneurs** en point à point avec un faible débit (commande de vitre, commande de rétroviseur, pilotage d'un alternateur): LIN : 19,2 kBits/s
- **Partage de données simples** n'ayant pas un fort besoin de synchronisation (régime moteur, vitesse de roue) pour des fonctions réparties entre calculateurs
(Sous-Capot, Habitacle):
 - >Habitacle CAN LS (250 kBits/s)
 - >Sous-Capot CAN HS (500 kBits/s)



11

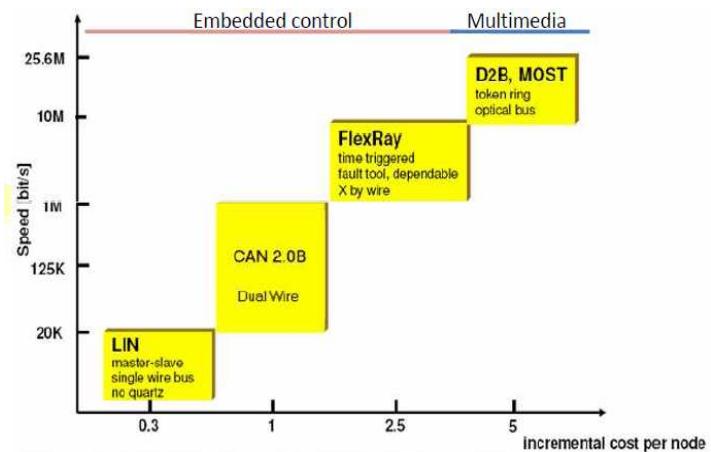
Introduction Générale (3/3)

- **Partage de données complexes** nécessitant d'être synchrone et demandant une forte disponibilité:
 - >Nouveau Réseau: FlexRay : 10 MBits/s
- **Partage de flux multimédia** dans l'habitacle:
 - >Nouveau Réseau: MOST: \approx 20 MBits/s
- NB: Tout ces réseaux répondent également à un besoin de robustesse à l'environnement automobile (CEM, T°C, vibrations)



12

Principaux réseaux automobiles



13

BUS CAN



14

HISTORIQUE

- 1980:Création du bus CAN par R.BOSCH
- 1987:Premiers Micro-contrôleurs CAN par Intel puis Philips)
- 1991:Normalisation du CAN Low-Speed, norme standard ISO 11519-1(250 Kbauds)
- 1992: DC implémente le CAN sur un véhicule
- 1993: Normalisation du CAN High-Speed, norme standard ISO 11898 (1Mbauds) (CAN 2.0 part A)
- 1995: Ajout du « CAN étendu » dans la norme ISO 11898 (CAN 2.0 part B)

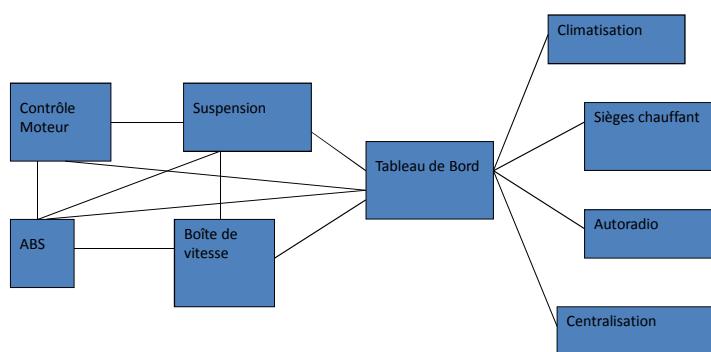


15

Le premiers type d'architecture de réseau

Réseau multiplexé constitué de long câbles :

- Type d'architecture véhicule vers la fin des années 80 et début 90
- Réseau conçu pour le multiplexage véhicule(conception BOSCH 1983- normalisation ISO 1994).



16

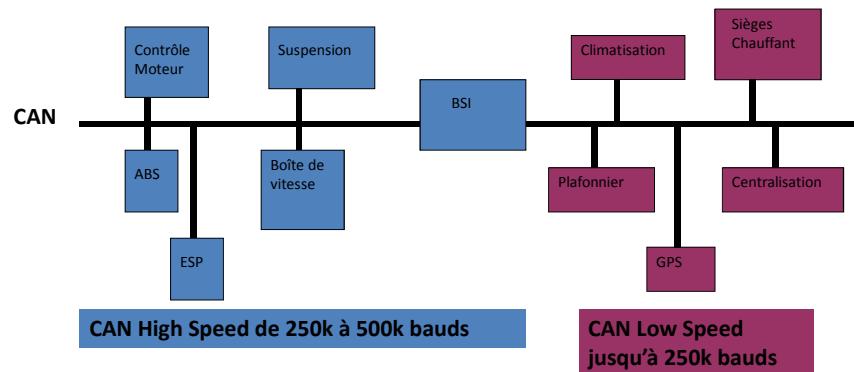
- Quelques chiffres:
 - Renault 25: 1 ECU et 2 km de câbles cuivre
 - Peugeot 407: 28 ECUs en 2004 et des dizaines de mètres de câbles
 - Mercedes Class E: > 40 ECUs en 2008 et des dizaines de mètres de câbles
 - Nouvelle série 7 de BMW : 82 ECUs pour toutes les options



17

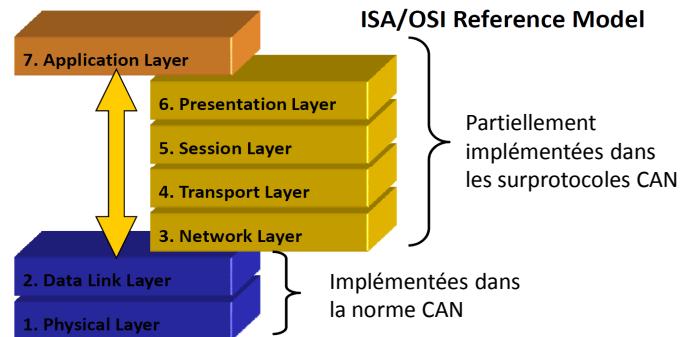
Type d'architecture de réseau CAN de nos jours

Réseau multiplexé constitué avec le bus CAN et quelques ECUs (PSA):



18

Le Modèle O.S.I pour le CAN



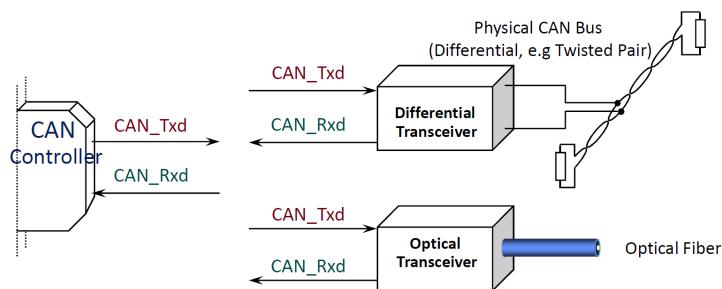
Pas de média physique définis dans la norme : obligation d'avoir un niveau récessif et un niveau dominant



19

Signaux

- Le media doit supporter un état dominant et un état récessif
- Media utilisé dans le monde de l'automobile
 - paire torsadée avec une résistance de terminaison de 120 ohms à chaque bout
 - signal différentiel(CAN_H, CAN_L)



20

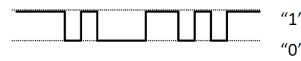
Dominant / Récessif

Caractéristiques Bus CAN

Deux états possibles sur le bus :

“1” = Etat récessif

“0” = Etat dominant



A	B	C	BUS
D	D	D	D
D	D	R	D
D	R	D	D
D	R	R	D
R	D	D	D
R	D	R	D
R	R	D	D
R	R	R	R

Dès qu'un nœud sur le bus transmet un bit dominant (zéro):
Le bus est dans un état dominant

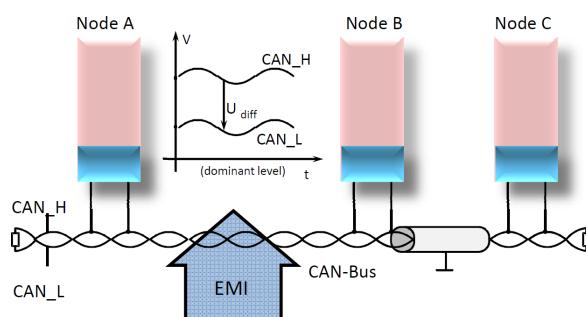
Si tous les nœuds transmettent un bit récessif (un):
Le bus est dans un état récessif

→ **Un bit dominant est prépondérant à un bit récessif sur le bus CAN**



21

Signaux électriques

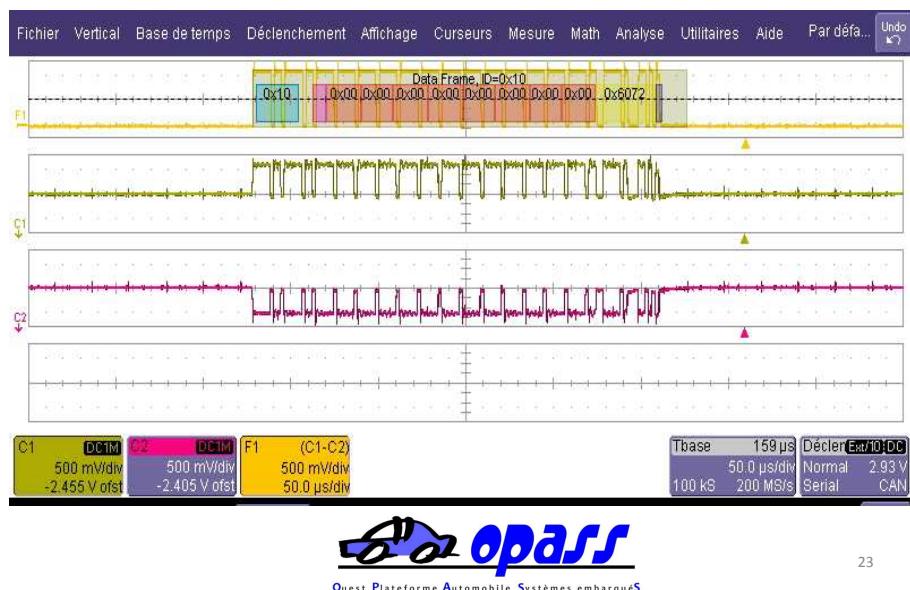


En cas de perturbation électromagnétique, les deux signaux sont perturbés de façon identique sans que l'on retrouve cette perturbation sur la différence



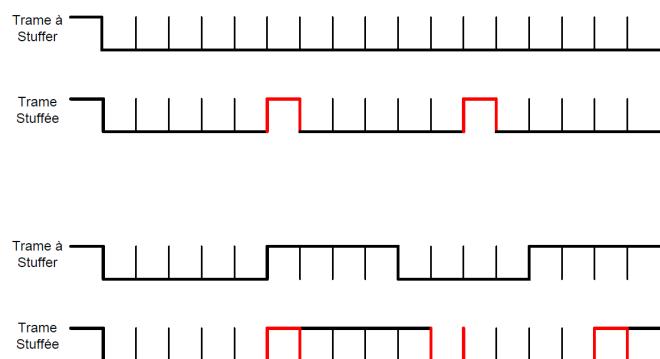
22

Signaux électriques



23

Bit stuffing



Cette méthode consiste, dès que l'on a émis 5 bits de même polarité sur le bus, à insérer un bit de polarité contraire pour casser des chaînes trop importantes de bits identiques.



24

Communication de type Event-Triggered:

=>Temps d'émissions individuelles

- Temps du départ et temps de latence inconnues à l'avance
- Nécessite un service d'arbitrage



25

Accès au BUS

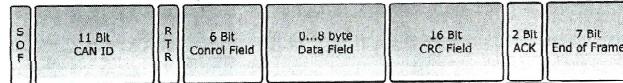
- Accès au BUS par multiplexage évenementiel via la Méthode CSMA/CA (Carrier Sense Multiple Access/Collision Advoidance)
- La collision de trame est non destructive
- L'arbitrage en CAN s'effectue uniquement sur l'identificateur et le bit RTR
- L'identificateur le plus petit gagne l'accès au bus



26

Le codage (1/2)

- Structure de la trame standard

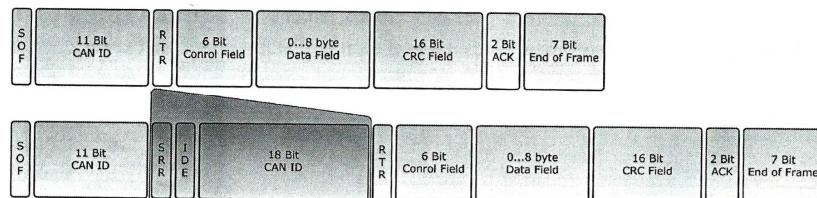


- SOF: marque le début de la trame 1 bit dominant
- Identifiant: 11 bits
- RTR: 1 bit
- Ctrl.: 6 bits dont 4 indiquent le nombre d'octet émis
- Informations: Data 8 octets maximum
- CRC: CRC sur 15 bits
- ACK: Acquittement sur 2 bits
- Fin: EOF sur 7 bits



27

Le codage (2/2)



Passage du CAN standard au CAN étendu

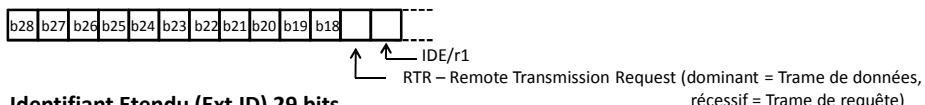
Les deux types de trames peuvent cohabiter sur le même réseau CAN



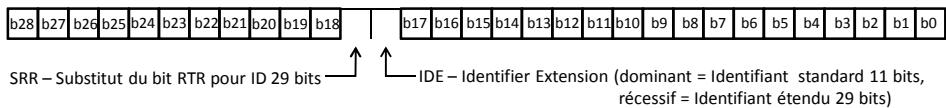
28

Identifiant (champ d'arbitrage)

Identifiant Standard (Std ID) 11 bits



Identifiant Etendu (Ext ID) 29 bits



Les bits dominants gagnants face au bits récessifs, les identifiants les plus petits sont plus prioritaires sur le bus. L'arbitrage entre une trame standard et une trame étendue se fait sur les 11 premiers bits de l'identifiant (b28 – b18). Si les identifiants sont identiques, la trame standard remporte l'arbitrage car son bit IDE est dominant.



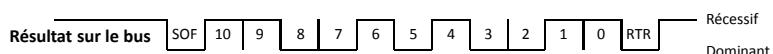
29

CSMA/CA(Carrier Sense Multiple Access/Collision Avoidance)

L'exemple suivant voit un module 1 envoyer un message avec l'identifiant 0x653, alors qu'au même moment un module 2 veut aussi envoyer un message mais avec l'identifiant 0x65B. Le module 2 perd l'arbitrage au bit 3. Le module 2 devra attendre que le bus soit de nouveau libre pour tenter d'envoyer sa trame



Le module 2 perd l'arbitrage et arrête de transmettre.



30

Trame de requête

-Une trame de requête est constituée de la même manière qu'une trame de données sauf que le champ de données est vide.

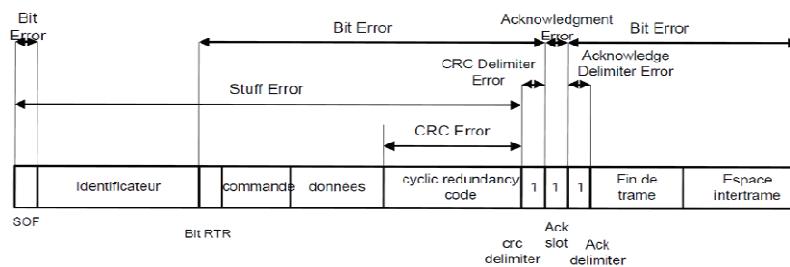
-Dans le champ d'arbitrage, le bit de RTR est récessif. Par conséquent si deux nœuds émettent chacun une trame possédant le même identificateur (c'est à dire qu'un nœud émet une trame de données et l'autre une trame de requête), l'arbitrage sur le bit de RTR va donner la priorité à la trame de données.

- Si un nœud a besoin d'un certain nombre de données, il va émettre une trame de requête dès que le bus sera libre en prenant soin d'indiquer dans le champ de contrôle le nombre d'octets de données dont il a besoin.



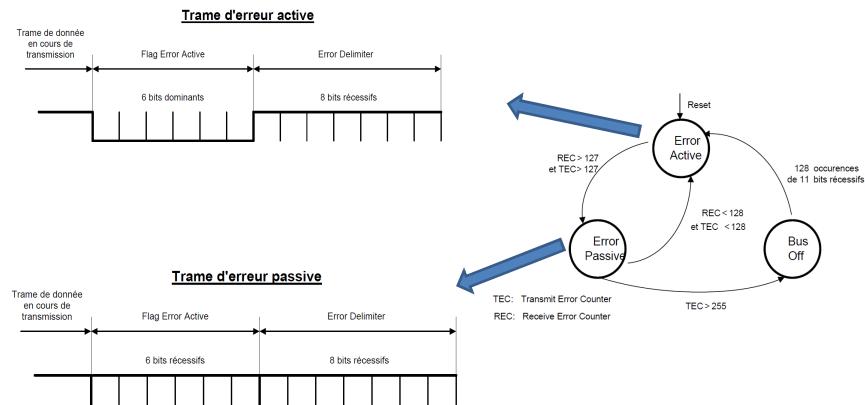
31

Gestion des erreurs (1/2)



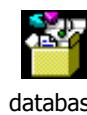
32

Gestion des erreurs (2/2)



33

Fichier Système CAN « .dbc »



database

34

Démonstration CAN



35

BUS J1939

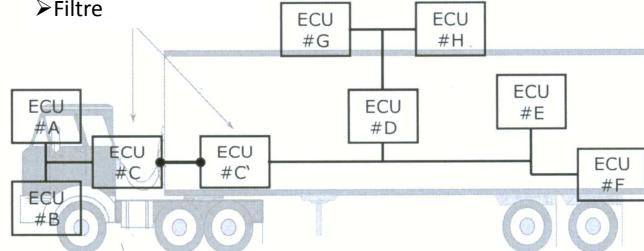


36

TOPOLOGY J1939

Pont / Passerelle

- 125 kbits/sec
- Filtre



Tracteur

- 250 kbits/sec
- Freins (ABS)
- Moteur
- ...

Remorque

- 250 kbits/sec
- Feux
- Freins
- ...



37

Ouest Plateforme Automobile Systèmes embarqués

NORMES

- SAE J1939 : Introduction au protocole
- SAE J1939/0x : Industry Group
 - J1939/01 : Camion et bus
 - J1939/02 : Équipement agricole
- SAE J1939/1x : Couche physique
 - J1939/11 : 250Kbits/s, Paire torsadée, blindée
 - J1939/12 : 250Kbits/s, 4 fils torsadés
 - J1939/13 : Prise diagnostique
- SAE J1939/21 : Protocole transport (DLC>8)
- SAE J1939/31 : Bridge, Router, Gateway, Filter
- SAE J1939/7x : Couche applicative
 - J1939/71 : Application commune
 - J1939/72 : Virtual Terminal
 - J1939/73 : Diagnostic
- SAE J1939/81 : Management de réseau (Assignation adresse dynamique)
- Le SAE J1939 peut être considéré comme le remplaçant du SAE J1708 et SAE J1587



38

Ouest Plateforme Automobile Systèmes embarqués

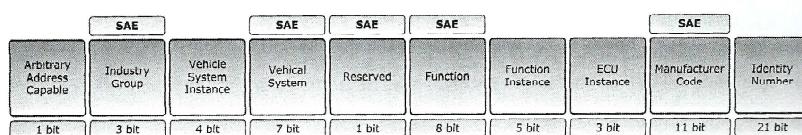
J1939

- Chaque module électronique posséde un nom unique (64 bit)
- Une adresse unique est attribuée pour chaque module dans un réseau spécifique (8 bit)
- Possibilité de faire une affectation dynamique des adresses à la mise sous tension ou par une commande via le gestionnaire de bus**



39

Nom du module (NMT)

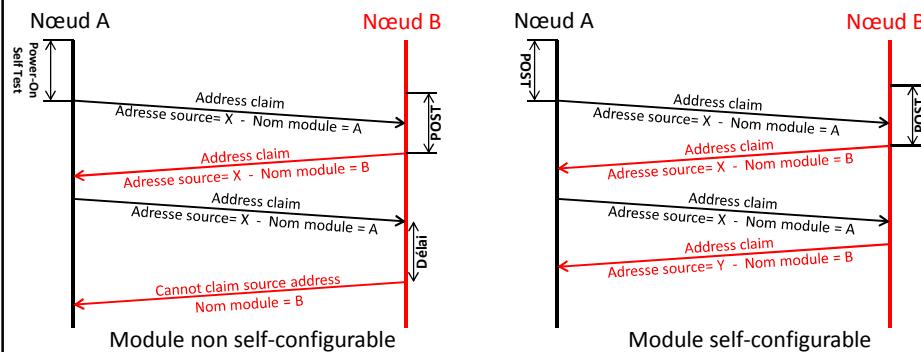


- Décrit la fonction du module
- Non unique dans le réseau
- Identifie le module et sera utilisé pour l'affectation des adresses dynamiques
- Chaque module envoyant un message doit avoir un nom
- Permet de résoudre des problèmes d'assignation d'adresses dynamique en définissant une priorité dans les modules



40

Adresse source des modules

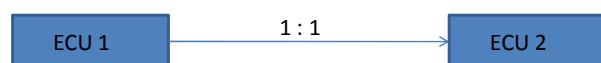


Possibilité de collision de trames, donc le SAE recommande
 « l'Address Claim Bus Collision Management »

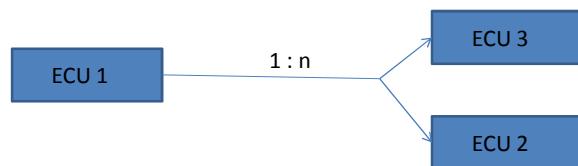


41

Types de communications



Destination spécifique (peer to peer)



Diffusion (broadcast)



42

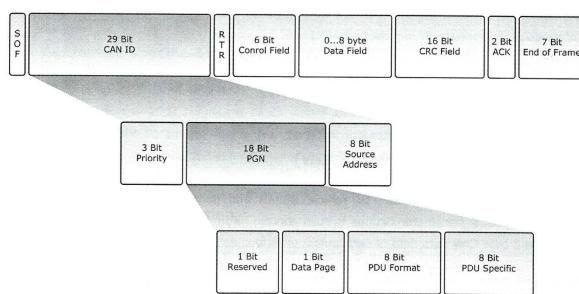
Parameter Group (PG)

- Description des données transmises par les modules
- Identificateur unique par le PGN
- Priorité fournie par le PG
- Pas limité aux 8 octets pour les datas
- Définition du type de transfert
- Data <= 8 octets → envoi d'un seul message
- Data > 8 octets → envoi de plusieurs messages



43

Parameter Group number (PGN)



	PDU Format	PDU Specific	Communication Mode
PDU1 Format	0 – 239 0hex - EFhex	Destination Address	Peer-to-Peer
PDU2 Format	240 – 255 F0hex - FFhex	Group Extension	Broadcasting



44

Exemple de SPN et PGN

(Suspect Parameter Number)

SPN 110

Engine Coolant Temperature

Temperature of liquid engine cooling system

Data Length 1 Byte

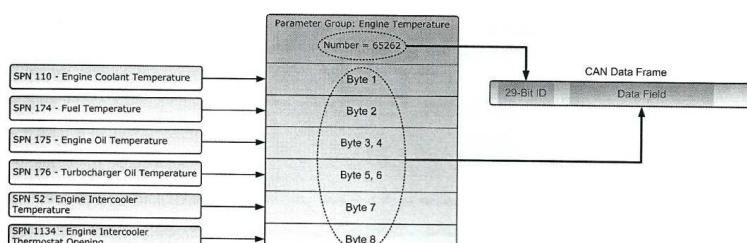
Resolution 1 deg C / Bit

Offset -40 deg C

Data Range -40 to 210 deg C

Type Measured

Reference



45

Type de trames

- Trame de commande
- Trame de requête
- Trame de broadcast /réponse
- Trame d'acknowledge
- Trame pour fonctions spécifiques (propriétaire)



46

Multi- Package

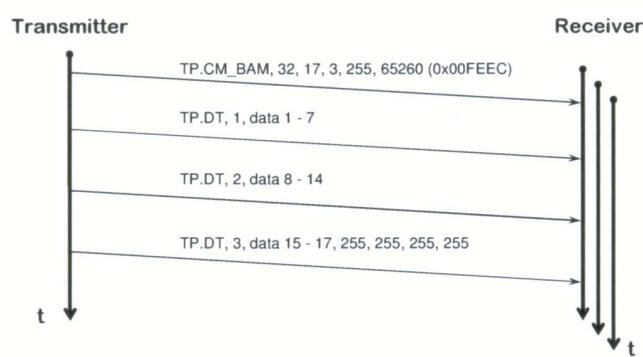
- Possibilité d'envoyer plus de 8 octets de data en utilisant le multi-package
- Envoi en broadcast ou en peer-to-peer
- Différents messages suivant le type de transmission:
 - . Broadcast Announce Message (BAM)
 - . Request to send
 - . Clear to send
 - . End of message acknowledgement
 - . Connection abort
 - . Data transfert



47

Exemple transfert 1 : n

Un nœud veut transmettre aux autres modules le message dont le PGN est 65260, ce message contient 17 octets de données



L'émetteur n'attend pas de réponse de la part des récepteurs, il attendra entre 50 et 200 ms entre chaque émission



48

Multi- Packet Peer-to-Peer

- L'envoi de plusieurs paquets en Peer-to-Peer se déroule en 3 étapes:
Connection Initialisation, Data Transfer, Connection Closure

- La Connection Initialisation consiste en l'envoi d'un Request to Send de la part de l'émetteur.
Le récepteur envoi, s'il est disponible, un Clear to Send.

- Le Request to Send et le Clear to Send sont contenus dans le Transport Protocol –
Connection Management (TP.CM) et a pour PGN 60416

Request to Send (TP.CM_RTS)

PDU Format =236

PDU Specific=Adresse récepteur

Priorité par défaut=7

Data:

1->Octet de contrôle=16

2,3->Taille du message en octets

4->Nombre total de paquets

5->Nombre max de paquets émis sans CTS

6-8->PGN du multi-packet message

Clear to Send (TP.CM_CTS)

PDU Format =236

PDU Specific=Adresse émetteur

Priorité par défaut=7

Data:

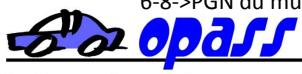
1->Octet de contrôle=17

2->Nombre paquets acceptés avant CTS
(≤ valeur de l'octet 5 du RTS)

3->Numéro du prochain paquet

4,5->Réservé (FF)

6-8->PGN du multi-packet message

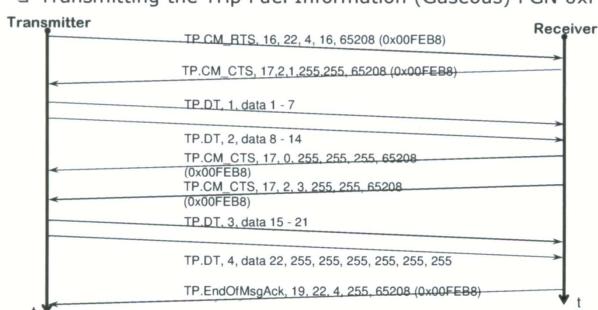


49

Exemple transfert 1 : 1

Un nœud veut transmettre aux autres modules le message dont le PGN est 65208,
ce message contient 22 octets de données

Transmitting the Trip Fuel Information (Gaseous) PGN 0xFEB8



CM_RTS: Control byte=16, PG size, total number of packets, default is 16, PGN of packaged message
CM_CTS: Control byte=17, Number of packets to send, Packet number to start with, reserved, reserved, PGN of packaged message
CM_EoAck: Control byte=19, PG size, total number of packets, reserved, reserved, PGN of packaged message
DT: Sequence number, Data bytes (up to 7 bytes available, not used data bytes are set to 255)



50

Trame de requête

- Le J1939 offre la possibilité à un module de demander à un nœud, ou à tous les nœuds, de lui envoyer un message spécifique

- Cette demande d'envoi de message est une requête (Request)

Request :

PGN 59904

Taille data = 8 octets

PDU Format = 234

PDU Specific = Adresse récepteur ou 255

Priorité par défaut = 6

Data:

1-3>PGN du message demandé

(Byte 1=PGN LSB et byte 3=PGN MSB)



51

Trame acknowledge

- Si un message est envoyé en Peer-to-Peer, il sera suivi nécessairement d'un acknowledge de la part du receveur

- Si un message est envoyé en Broadcast, aucun acknowledge n'est nécessaire

- Le J1939 regroupe deux types d'acknowledge:

- L'acknowledge classique du CAN, qui n'est pas la confirmation de la bonne réception du message par le destinataire du message, mais la confirmation de la bonne réception du message par un nœud du réseau (calcul du CRC).

- Les trames d'acknowledgement J1939, qui sont la confirmation de la réception d'un message spécifique



52

Trame acknowledge

Acknowledgement :

PGN 59392

Taille data = 8 octets

PDU Format = 232

PDU Specific = Adresse récepteur ou 255

Priorité par défaut = 6

Data:

1-8> Positive Acknowledgement,

Négative Acknowledgement,

Access Denied

Cannot Respond



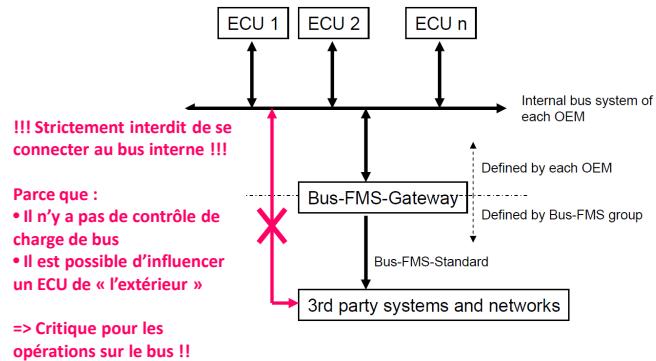
53

BUS FMS



54

Bus FMS



55

Bus FMS

- FMS autobus ou FMS camion (indiqué dans la trame FMS-standard interface)

- Connecteur normalisé pour les autobus

Daimler-Benz-Electro-Cable		Sonus CV	Siemens Ifaco	UAW-Standard	Bus FMS-Standard	Ref. SAE J1939
MAN Nutzfahrzeuge AG		Value Bus Corporation	VRL Bus International B.V.	UAW-Standard	Bus FMS-Standard	Ref. SAE J1939
VDO AG		ACEA Working Group BCEU		UAW-Standard	Bus FMS-Standard	Ref. SAE J1939
Date:				07/07/2000	Version	Rev. 00.02
Code:					Reg. no.	
Bus FMS-Standard interface description according SAE J1939						
3 Connector description						
DIN 72585 connector 4-Pin male type (vehicle side)						
AMP						
Chassis part (vehicle)		Coding 1 (black)				
Housing w/o pincontact housing		1947401				
Housing w/o pincontact housing		1947401	1947401			
Pin (in plate)		1952588	1952588			
Gel		039501	039501			
Cap (for not used connector)		1946272	1946272			
AMP						
Housing		1947401				
Housing		1947401				
Socket (in plate)		039501				
Gel		039501				
GH						
Chassis part (vehicle)						
Housing		19387200200				
Pincontact housing		1952588	1952588			
Pin (in plate)		1952588	1952588			
Gel		1944164	1944164			
Cap (for not used connector)		N/A				
GH						
FMS side (male)						
Housing		17964000302				
Pincontact housing		1952588	1952588			
Pin (in plate)		1952588	1952588			
Gel		1944164	1944164			
ORIENTATION R/P						
Pin Layout:						
Pin 1 CAN high						
Pin 2 CAN low						
Pin 3 Option CAN ground						
Pin 4 not used by Bus-FMS-Standard						



56

Infos disponibles sur FMS camion

PGN	Description	Paramètres	SPN
65265	Régulateur vitesse/ Vitesse véhicule (CCVS)	Vitesse des roues	84
		Etat du régulateur de vitesse (Activé ou non)	595
		Position pédale de frein	597
		Position pédale d'embrayage	598
		Etat PTO (Statut du conducteur)	976
61443	Contrôleur moteur électronique #2	Position pédale accélérateur	91
65257	Consommation carburant (LFC)	Carburant total utilisé	250
65276	Ecran tableau de bord (DD)	Niveau carburant restant	96
61444	Contrôleur moteur électronique #1	Vitesse rotation moteur	190
65258	Poids véhicule (VW)	Numéro pneu sur l'essieu en comptant de gauche à droite	928
		Numéro essieu en comptant de l'avant vers l'arrière	928
		Poids exercé sur l'essieu	582
65253	Nombre total d'heure du moteur	Nombre d'heure d'utilisation du moteur depuis réinitialisation	247
65260	Identification véhicule (VI)	Numéro d'indentification du véhicule (VIN)	237
64977	Identification interface FMS/ Capacités(FMS)	Trame Diagnostic supportées (O/N)	2804
		Trame Requête supportées (O/N)	2805
		Version Software supportée	2806



57

Infos disponibles sur FMS camion

PGN	Description	Paramètres	SPN
65217	Distance précise parcourue (VDHR)	Distance totale parcourue par le véhicule	917
65216	Service (SERV)	Distance restante avant la prochaine révision	916
65132	Tachograph (TCO1)	Etat travail conducteur 1	1612
		Etat travail conducteur 2	1613
		Etat véhicule (Arrêt ou déplacement)	1611
		Temps de travail conducteur 1	1617
		Présence ou non de la carte conducteur 1	1615
		Vitesse excessive	1614
		Temps de travail conducteur 2	1618
		Présence ou non de la carte conducteur 2	1616
		Détection d'événement tachygraphe	1622
		Manipulation d'information	1621
		Performance tachygraphe	1620
		Direction du véhicule (Marche avant ou arrière)	1619
		Vitesse véhicule tachygraphe	1624
65262	Température moteur	Température liquide de refroidissement	110



58

Infos disponibles sur FMS autobus

PGN	Description	Paramètres	SPN
65265	Régulateur vitesse/ Vitesse véhicule (CCVS)	Vitesse des roues	84
		Etat du régulateur de vitesse (Activé ou non)	595
		Position pédale de frein	597
		Position pédale d'embrayage	598
61443	Contrôleur moteur électronique #2	Position pédale accélérateur	91
65257	Consommation carburant (LFC)	Carburant total utilisé	250
65276	Ecran tableau de bord (DD)	Niveau carburant restant	96
61444	Contrôleur moteur électronique #1	Vitesse rotation moteur	190
65253	Nombre total d'heure du moteur	Nombre d'heure d'utilisation du moteur depuis réinitialisation	247
65260	Identification véhicule (VI)	Numéro d'identification du véhicule (VIN)	237
64977	Identification interface FMS/ Capacités(FMS)	Trame Diagnostic supportées	2804
		Trame Requête supportées	2805
		Version Software supportée	2806
65217	Distance précise parcourue (VDHR)	Distance totale parcourue par le véhicule	917



59

Infos disponibles sur FMS autobus

PGN	Description	Paramètres	SPN
65132	Tachograph (TCO1)	Etat travail conducteur 1	1612
		Etat travail conducteur 2	1613
		Etat véhicule (Arrêt ou déplacement)	1611
		Temps de travail conducteur 1	1617
		Présence ou non de la carte conducteur 1	1615
		Vitesse excessive	1614
		Temps de travail conducteur 2	1618
		Présence ou non de la carte conducteur 2	1616
		Détection d'événement tachygraphe	1622
		Manipulation d'information	1621
		Performance tachygraphe	1620
		Direction du véhicule (Marche avant ou arrière)	1619
		Vitesse véhicule tachygraphe	1624
65262	Température moteur	Température liquide de refroidissement	110
65265	Régulateur vitesse/Vitesse véhicule	Etat frein parking	70
65269	Conditions ambiantes (AMB)	Température de l'air ambiant	171
65102	Contrôle portes 1 (DC1)	Position des portes	1821
		Rampe/utilisation fauteuil roulant	1820
		Statut 2 des portes	3411
64933	Contrôle portes 2 (DC2)	Statut fermeture porte 1	3412
		Statut ouverture porte 1	3413
		Autorisation statut porte 1	3414
		Statut fermeture porte 2	3415



60

Infos disponibles sur FMS autobus

PGN	Description	Paramètres	SPN
65254	Heure/ Date (TD)	Secondes	959
		Minutes	960
		Heures	961
		Jour	962
		Mois	963
		Année	
65198	Pression d'air (AIR1)	Pression circuit d'air du frein de service #1	1087
		Pression circuit d'air du frein de service #2	1088
65237	Vitesse alternateur (préliminaire) (AS)	Statut alternateur 1	3353
		Statut alternateur 2	3354
		Statut alternateur 3	3355
		Statut alternateur 4	3356
61445	Contrôleur électronique de transmission 2 (ETC2)	Vitesse sélectionnée	524
		Vitesse engagée	523
61445	Contrôle de suspension pneumatique (ASC4)	Pression sur l'essieu avant gauche	1725
		Pression sur l'essieu avant droit	1726
		Pression sur l'essieu arrière gauche	1727
		Pression sur l'essieu arrière droit	1728
65131	Identification conducteur (DI)	Identification conducteur 1	1625
		Identification conducteur 2	1626
65266	Economie de carburant (LFE)	Débit de carburant	182
		Economie carburant instantanée	184
64893	FMS Tell Tale Status	Telltale Identification bloc	
		Telltale Status 1-14	



61

Démonstration bus J1939



62

BUS ISO11783 / ISOBUS

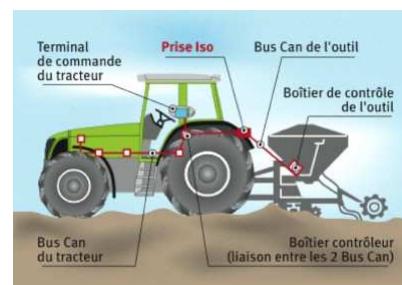


63

Isobus ou ISO11783

Dans le cas d'un pilotage d'outil conventionnel, le calculateur est intégré dans le boîtier de commande de l'outil placé en cabine. Sur l'outil, on ne retrouve que des éléments ayant pour seule fonction d'exécuter les ordres reçus du boîtier de commande dans lequel se trouve toute la programmation. De ce fait, chaque outil nécessite son propre boîtier.

Avec l'Isobus, les ordres donnés par le chauffeur au niveau du terminal unique en cabine sont transférés à l'outil par une seule connexion. Ces ordres sont retravaillés et convertis au niveau de l'outil. Le boîtier de commande est placé sur l'outil, le terminal en cabine ne sert alors qu'à la visualisation et à la rédaction de l'ordre, un peu comme un clavier et un écran pour un ordinateur. Le terminal en cabine s'adapte ainsi à n'importe quel outil compatible Isobus en chargeant les données de configuration de l'écran à partir du boîtier embarqué sur l'outil.



64

Isobus ou ISO11783

-ISO 11783 / ISOBUS est basé sur la norme SAE J1939 (Truck and bus control and communications network subcommittee)

-Définit la couche physique (1), liaison (2), transport (4) et application (7) du modèle OSI

-Est similaire au bus J1939 pour certains détails du protocole tout en intégrant quelques différences.

-Utilise les identifiants étendus (29 bits)



65

Couche applicative (7)

- Virtual Terminal
- Task controller
- Tractor ECU
- File server

La couche applicative définit comment doivent fonctionner ces modules et les informations qu'ils peuvent recevoir et émettre sur le bus.



66

Virtual terminal

Le concept du terminal virtuel est :

- Un module transmet par le CAN une définition de ce qui sera à afficher sur le terminal. Cela s'appelle un « object pool »
- « l'object pool » définit les couleurs, formes, images, données d'entrées et de sorties que le terminal virtuel devra afficher
- Le terminal virtuel sait interpréter l'object pool et sait comment afficher les différentes infos.
- Le terminal virtuel ne connaît pas le fonctionnement des modules dont il va devoir afficher les informations
- Si l'utilisateur manipule le terminal virtuel, les données de sorties seront envoyées au module par le bus CAN



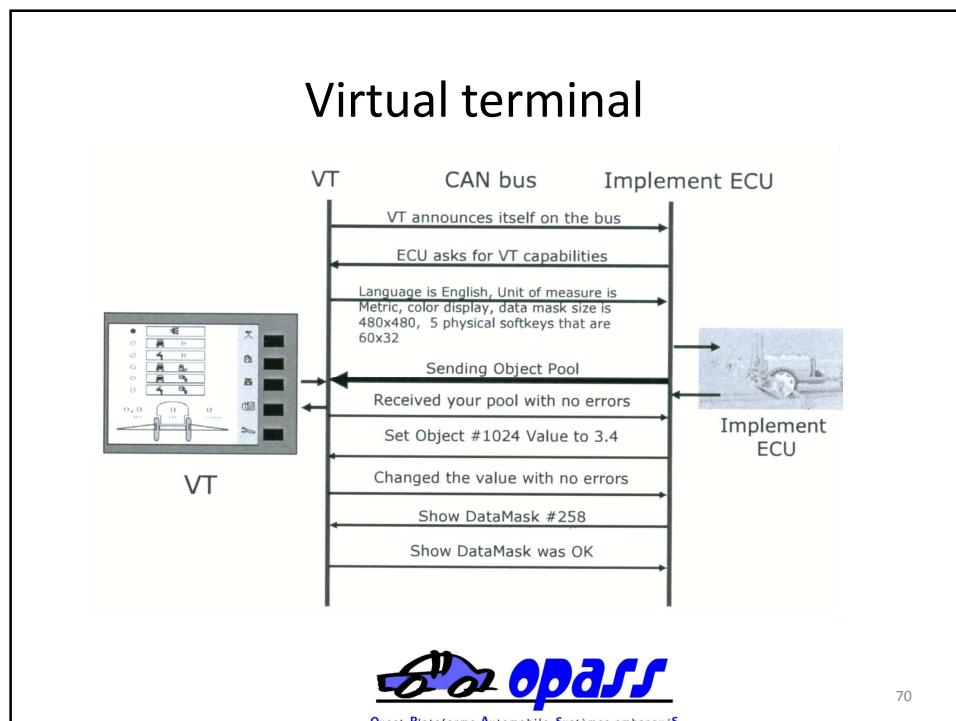
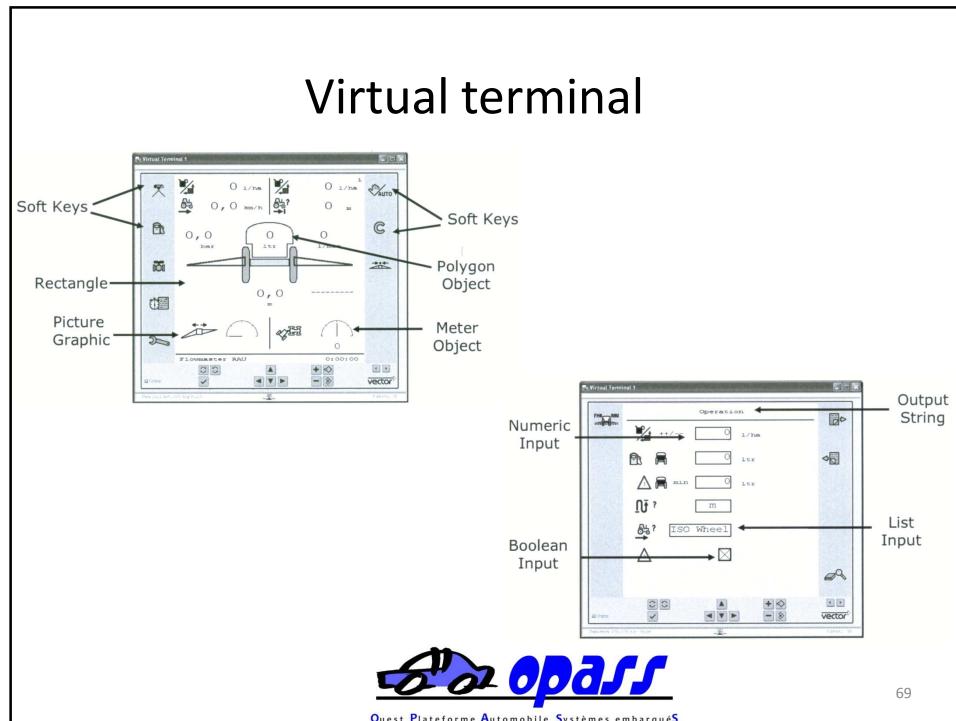
67

Virtual terminal

- ❑ Object Pools are comprised of 32 objects
 - ❑ Working Set Object
 - ❑ Data Mask Object
 - ❑ Alarm Mask Object
 - ❑ Container Object
 - ❑ Soft Key Mask & Soft Keys
 - ❑ Button Object
 - ❑ Input Field Objects
 - ❑ Boolean
 - ❑ String
 - ❑ Number
 - ❑ List
 - ❑ Output Field Objects
 - ❑ String
 - ❑ Number
 - ❑ Output Shape Objects
 - ❑ Line
 - ❑ Rectangle
 - ❑ Ellipse
 - ❑ Polygon
 - ❑ Output Graphic Objects
 - ❑ Meter
 - ❑ Bar Graph
 - ❑ Arched Bar Graph
 - ❑ Picture Graphic Object
 - ❑ Variable Objects
 - ❑ String
 - ❑ Number
 - ❑ Attribute Objects
 - ❑ Font
 - ❑ Line
 - ❑ Fill
 - ❑ Input
 - ❑ Pointer Object
 - ❑ Macro Object
 - ❑ Auxiliary Objects
 - ❑ Auxiliary Input
 - ❑ Auxiliary Function
 - ❑ Auxiliary Assignment (Proposed)



68



Task Controller

- Pas obligatoire, mais conseillé car le bus permet d'automatiser facilement les tâches
- Fait le lien entre le PC de l'agriculteur et le système embarqué
- Décrit les différents systèmes embarqués
- Décrit le process
- Le Task Controller peut être un module à part entière ou être intégré à un autre module.



71

Tractor ECU (TECU)

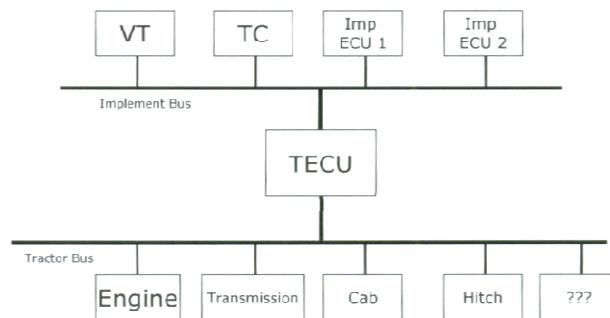
- Module électronique qui réalise la passerelle entre le bus tracteur (CAN ou autre) et le bus ISOBUS.
- Permet aux modules ISOBUS d'accéder aux données et fonctionnalités du tracteur
- Permet de gérer la partie puissance fournie aux modules ISOBUS
- Il existe trois classes de TECU, dépendant du nombre d'information du tracteur pouvant être transmis sur l'ISOBUS



72

Tractor ECU (TECU)

□ Gateway Functionality



73



Ouest Plateforme Automobile Systèmes embarqués

File server

-Module qui permet de sauvegarder des fichier de données

-Possibilité de faire :

- Lecture
- Ecriture
- Suppression des fichiers
- Renomage des fichiers
- Attributs des fichiers
- Directory

-Utilise deux PGNs

- File server to client (AB00)
- Client to file server (AA00)



74

Ouest Plateforme Automobile Systèmes embarqués

Démonstration ISOBUS



75

BUS NMEA2000



76

NMEA 2000

- Successeur du NMEA 0183 (liaison série)
- Compatibilité indiqué par : 
- Permet l'interopérabilité des équipements
- Définit le type de câbles et de connecteurs
- Norme IEC 61162-3



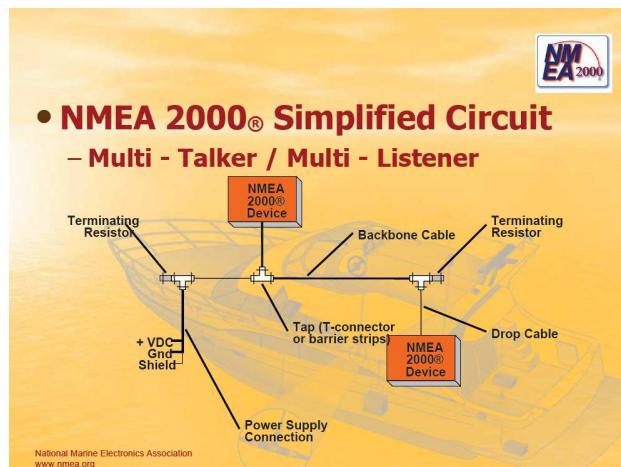
77

NMEA 2000



78

NMEA 2000



79

NMEA 2000

- Basé sur le bus J1939

- Vitesse : 250kbits/sec

- Database spécifique



Nmea2000dbc



80

BUS CANopen



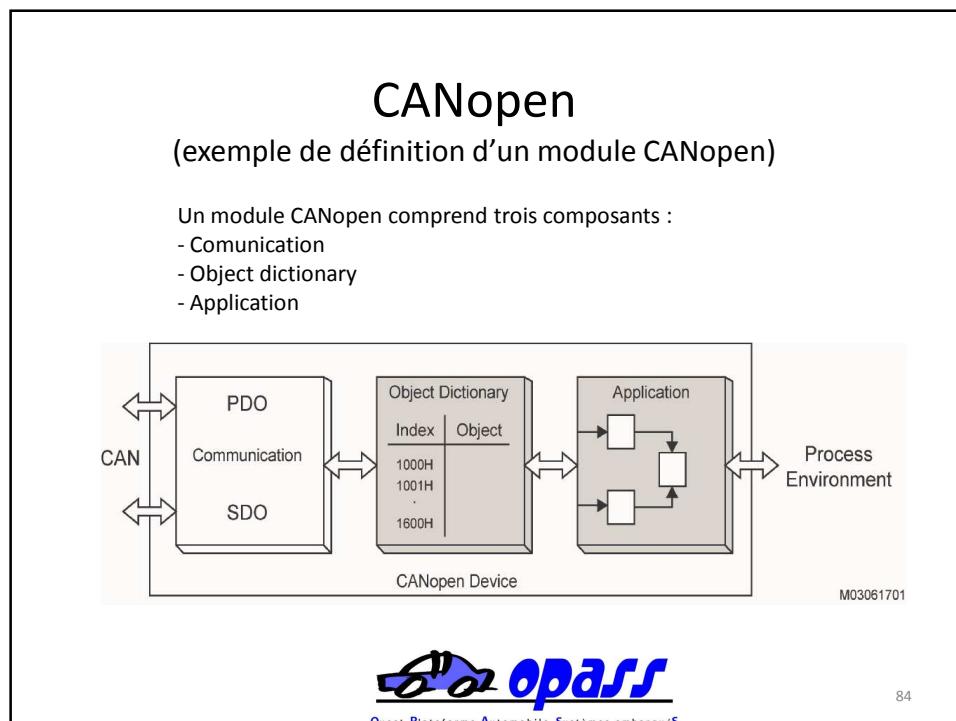
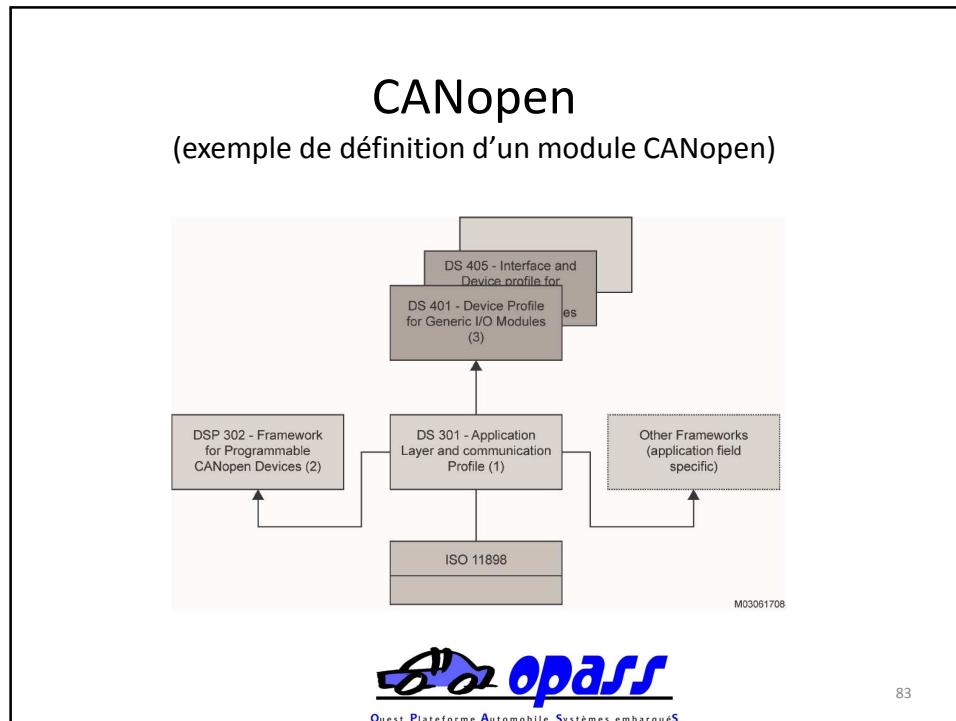
81

CANopen

- Utilisé dans l'industrie
- Couche 7 (applicative) du modèle OSI
- Identifiants 11 bits
- Définition de profils
- Définition de l'adresse des modules par hardware



82



CANopen

(Object dictionary - OD)

Index (hex)	Object
0000	not used
0001-025F	Data Types
0260-0FFF	Reserved for further use
1000-1FFF	Communication Profile Area
2000-5FFF	Manufacturer Specific Profile Area
6000-9FFF	Standardized Device Profile Area
A000-BFFF	Standardized Interface Profile Area
C000-CFFF	Reserved for further use



85

CANopen

(mode de communication)

- Client / serveur :
 - Communication point à point
 - N clients, n serveurs
- Maître / esclave :
 - Communication point à point
 - 1 maître, 1...127 esclaves
- Producteur / consommateur
 - Communication « broadcast »
 - N producteurs, n consommateurs



86

CANopen

(Process Data Object - Service Data Object)

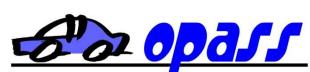
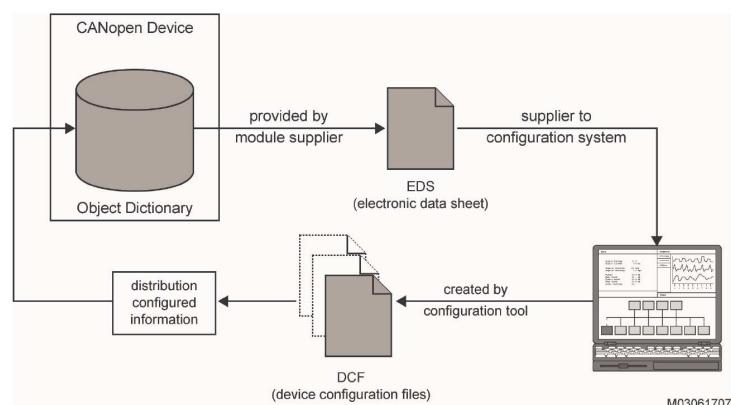
Process Data Object (PDO)	Service Data Object (SDO)
Priorité haute (échange temps réel)	Priorité basse
Sans acknowledge	Avec Acknowledge
Mode broadcast	Communication point à point
Max 8 octets de data	Pas de limites
Mode synchrone, asynchrone, cyclique ou événementiel	
process	Configuration et diagnostique



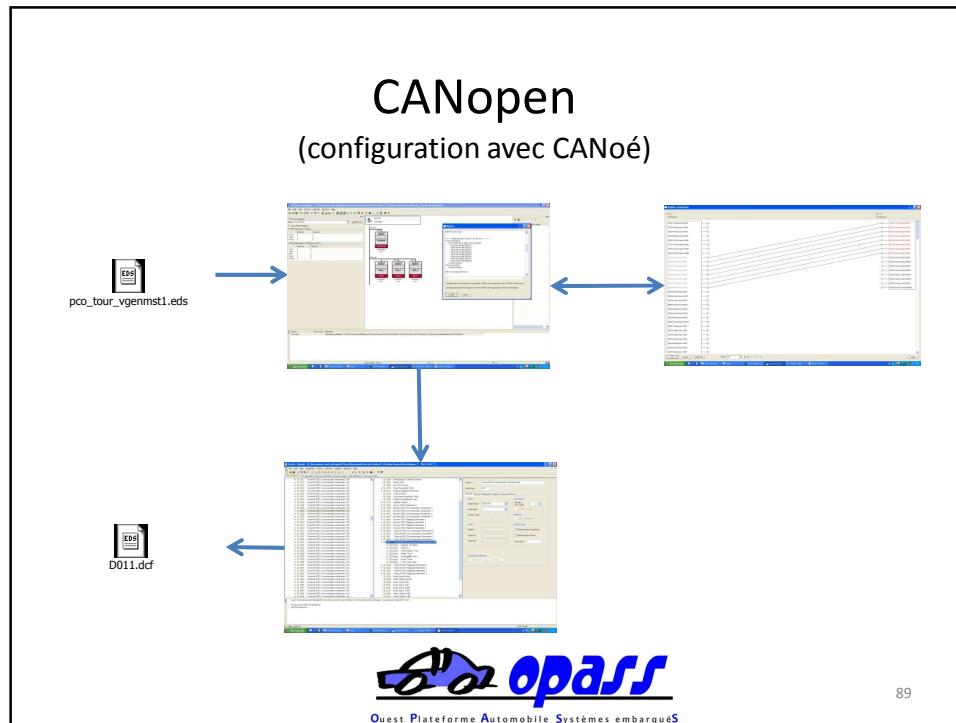
87

CANopen

(Configuration du réseau)



88



Objectifs

- Définir un standard de multiplexage ouvert pour un sous bus CAN
- Facile à utiliser (principe de l'UART) et de bonne fiabilité
- Garantie des temps d'attente pour un signal transmis
- D'un coût deux à trois fois moins cher que le CAN
- Aucune licence pour les membres



91

Standard LIN

Standard LIN décrit :

- Le protocole de transmission (LIN Protocol Specification)
- Le medium de communication (couche physique)
- L'interface entre les outils de développement (LIN configuration, language description et LIN API)
- L'interface des logiciels de programmation



92

Standard LIN

Version	Date
Lin 1.0	01/07/1999
Lin 1.1	06/03/2000
Lin 1.2	17/11/2000
Lin 1.3	13/12/2002
Lin 2.0	06/09/2003
Lin 2.1	24/11/2006



93

Couche physique

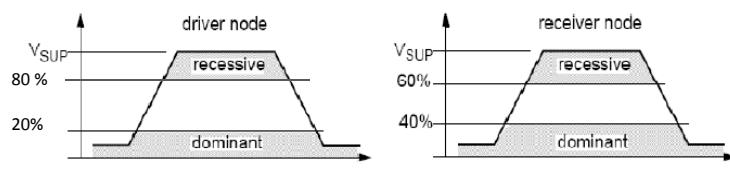
- Le lin est basé sur la couche physique ISO9141 (K line)
- Sa vitesse est variable de 1 à 20kbit/s
- La transmission est sur un fil
- Il utilise l'interface standard série (Uart des microcontrôleurs) contrairement au bus CAN qui utilise une interface spécifique.
- Codage type 8N1 (start bit, 8 databit, stop bit)
- Le bus à une longueur maximum de 40 m.



94

Couche physique

- Le medium est un simple fil connecté à l'alimentation par une résistance de pull-up sur chaque nœud.
- La diode en série protège d'une éventuelle perte de tension batterie.



95

Technique de transmission

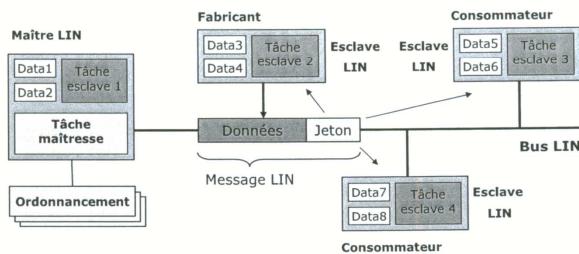
- Transmission en simple maître/multi-esclaves.
- Les données utiles sont de 1 à 8 Octets
- La topologie est un bus unifilaire.
- Longueur maxi 40m et des résistances de fin de ligne
- Vitesse pouvant aller jusqu'à 20 kbit/s
- Le nombre de nœuds max est 16 (1 maître et 15 esclaves maximum).



96

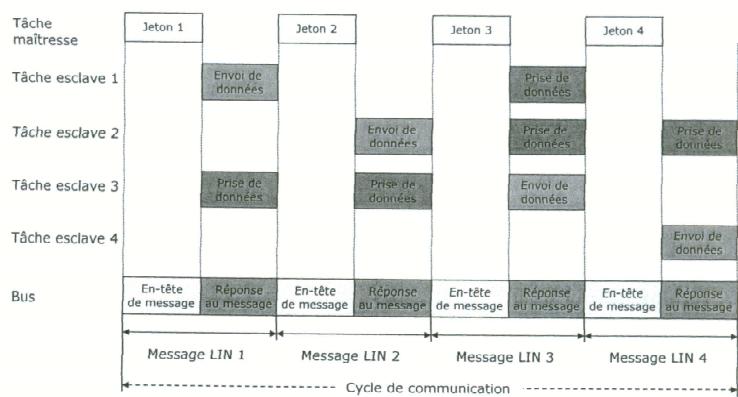
Communication par jeton (1/2)

- ❑ Les nœuds LIN ne disposent pas du même accès au bus en raison de l'architecture maître-esclave
 - ❑ Le maître LIN délègue la communication (principe du jeton délégué)
 - ❑ Distribution des messages basée sur l'adresse des messages
 - ❑ 64 adresses de message (identificateurs)



97

Communication par jeton (2/2)



98

Prédicibilité

- Approche Time triggered:
 - Longueur du message est connue.
 - La longueur minimum peut être calculée
 - Chaque message a une réserve de 140% de sa longueur minimum.
 - Longueur maximale permise est connue
 - Distance entre le début de deux messages
- Le séquencement des messages est connu
 - Le maître utilise un enchainement programme des envois :
 - mes1, mes2, mes3, mes1, mes4, mes5, mes1, ...
 - Il peut y avoir plusieurs tables programmées
- L'utilisation de table de programmes fournit une certaine flexibilité



99

Format des messages

- Il n'y a pas de notion d'adresse explicite mais d'identificateur qui est le seul filtrage des messages pour les esclaves.

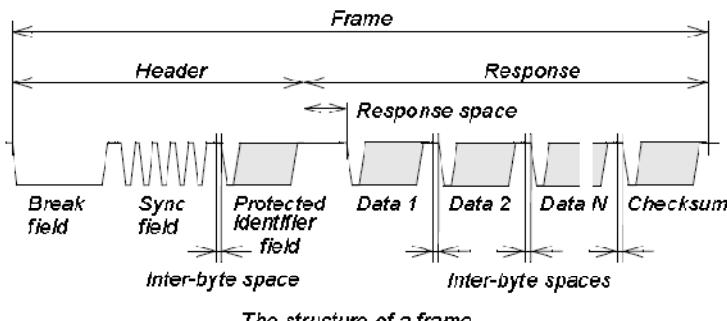
Couches ISO

- Couche 1 : physique, accès medium
- Couche 2 : transfert, communications de données avec gestion
- Couche 3 à 6 inutilisées



100

Format des messages



101

Composition de la trame

Composition d'une trame maître

- Temps mort de synchronisation (synch break) : 13 bits mini.
- Champs de synchronisation (sync field) : 1 octet
- Champ identificateur (identifier field) : 1 octet

Composition d'une trame esclave en réponse

- Champ de données : 1 à 8 octets de données
- Champ de contrôle : 1 octet



102

Composition de la trame

Champ break de synchronisation

- Le break de synchronisation fournit un reset pour démarrer la synchronisation des esclaves ayant une horloge imprécise.
- Il est composé de 13 bits dominants transmis par le maître uniquement suivis par un délimiteur break d'un bit récessif.
- Il sert de message de début de trame pour tous les nœuds du bus

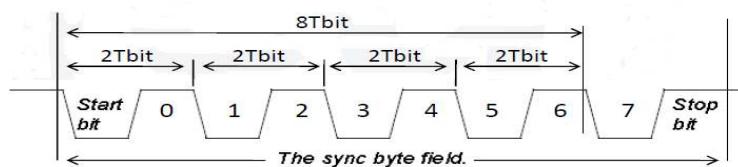


103

Composition de la trame

Champ de synchronisation

- Tbit dépend du maître, il n'y a pas de segmentation du bit comme dans le CAN, c'est l'horloge du maître qui cadence le bus. La synchronisation est en fait par une trame envoyée par le maître.
- Cette trame de synchronisation est 0x55 et on mesure le temps entre le front descendant du start bit ou fin de trame de break et le front descendant du bit 7 et on divise par 8, on obtient le Tbit.

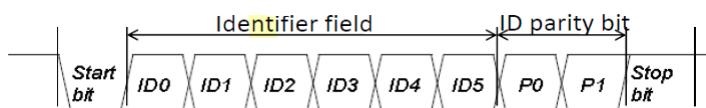


104

Composition de la trame

Champ identificateur

- ID0-5 identifiant du message :
 - 0-63 : plage d'adresse
 - 60,61 : diagnostic
 - 62,63 : réservé
 - ID6-7 la parité : P0 indique la parité paire et P1 la parité impaire.
- > $P0 = ID0 \oplus ID1 \oplus ID2 \oplus ID3$
 -> $P1 = \text{INV}(ID1 \oplus ID3 \oplus ID4 \oplus ID5)$



Mapping of frame identifier and parity to the protected identifier byte field.



105

Composition de la trame

Champ données

- Envoyé par l'esclave
- La longueur des données est de 1 à 8 octets.
- La transmission commence par le LSB.



Numérotation des octets de données
dans une trame contenant huit octets de données

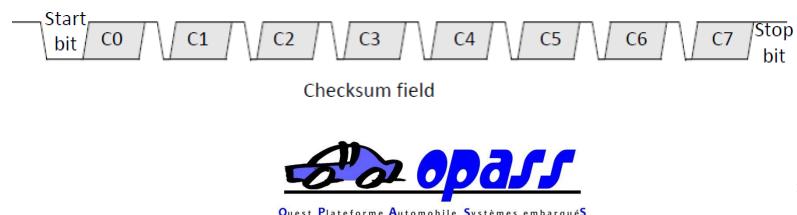


106

Composition de la trame

Champ de contrôle

- Checksum standard calculé sur le champ de données (LIN 1.x).
- Checksum amélioré sur champ de données et champ d'identifiant (LIN 2.x)
- Les messages avec identifiants 60 et 61 (diagnostiques) sont toujours avec un checksum standard sur le champ de données.



107

Trame de mise en veille (sleep)

La mise en mode veille est pilotée par une trame maître avec le premier champ de donne a 0 puis les autre a 0xFF. L'esclave devra ignorer les champs de données de 2 a 8 et analyser que le premier champ de donne.

Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7	Data 8
0	0xFF						

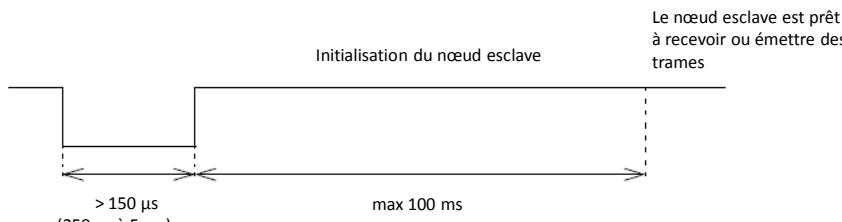
Commande de mise en veille



108

Trame de réveil

- Tous les nœuds en mode veille peuvent envoyer une requête de réveil (wake-up). Ils doivent tous pouvoir détecter une requête de réveil.



109

Gestion des erreurs

5 mécanismes de détection d'erreurs sur les données et l'identificateur:

- Erreur de bit (maître/esclaves)
- Erreur d'identificateur détectée par les parités (maître/esclaves)
- Erreur de champ de synchronisation, un esclave n'a pas détecté dans le temps maxi prévu (oscillateur des esclaves)
- Erreur de non réponse de l'esclave (maître)
- Erreur checksum (maître)

- Le mécanisme de correction d'erreurs ne fait pas partie du protocole LIN comme pour le CAN mais des recommandations sont données pour leurs gestions.



110

Comparaison avec le CAN

	LIN	CAN
Interface	UART/SCI	Spécifique CAN
Topologie	Bus unifilaire (V Bat) 40m	Bus bifilaire différentiel 5 ou 12v 1000 à 40m
Vitesse (kbits/s)	1 à 20	125 à 1000
Mode d'adressage	Identificateur	Identificateur
Nombre maxi de noeuds	16	20-30 pour 2032 identificateurs en CAN 2A
Efficacité de la trame (bits utiles/bits totaux)	30 à 60% (2 à 8 octets de données)	0 à 53% (0 à 8 octets de données)
Couche physique	ISO9141	ISO11898
Sécurité	Moyenne	Elevée (fault tolerant)
Mode d'accès au réseau		Arbitrage bitwise CSMA/CD
Hiérarchie entre équipements	Procédure Maître/Escalve	Multi-maître
Logiciel	Simple interface série	Couche application selon
Coût par noeuds	Environ \$1	Environ \$3
Interface standard	API, Couche Physique LLC, MAC	Couche Physique LLC, MAC



111

Démonstration bus LIN



112

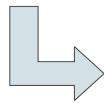
Limites du CAN



113

Limites du CAN (1/5) calcul des pires temps de réponses

	10 ms	20 ms	50 ms	100 ms
Ordinateur de Bord	-	2 Trames	4 Trames	4 Trames
Calculateur Contrôle Moteur	2 Trames	2 Trames	1 Trame	1 Trame
Calculateur Frein	1 Trame	1 Trame	-	2 Trames
Capteur d'Angle Volant	1 Trame	1 Trame	-	2 Trames
Accéléromètre	-	1 Trame	-	2 Trames
Calculateur Boite de Vitesse	1 Trame	1 Trame	-	2 Trames
Calculateur de Suspension	-	1 Trame	-	2 Trames
Calculateur de Direction Assistée	-	1 Trame	-	2 Trames



5 Trames à 10 ms
10 Trames à 20 ms
5 Trames à 50 ms
17 Trames à 100 ms

Hypothèses retenues:

- Débit 500 kbits/s
- CAN 2.0B
- Trame de 8 octets



114

Limites du CAN (2/5) calcul des pires temps de réponses

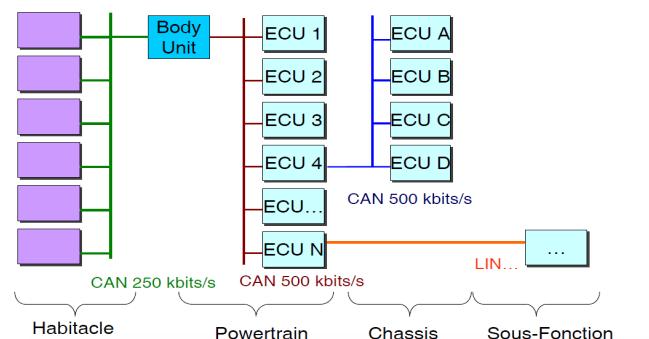
ID	Période	Tframe	Tretard	%delais
0x001	10 ms	262 µs	262 µs	2,616%
0x002	10 ms	262 µs	523 µs	5,232%
0x003	10 ms	262 µs	785 µs	7,848%
0x004	10 ms	262 µs	1046 µs	10,464%
0x005	10 ms	262 µs	1308 µs	13,08%
0x006	20 ms	262 µs	1570 µs	7,848%
0x007	20 ms	262 µs	1831 µs	9,156%
0x008	20 ms	262 µs	2093 µs	10,464%
0x009	20 ms	262 µs	2354 µs	11,772%
0x010	20 ms	262 µs	2616 µs	13,08%
0x011	20 ms	262 µs	2878 µs	14,388%
0x012	20 ms	262 µs	3139 µs	15,696%
0x013	20 ms	262 µs	3401 µs	17,004%
0x014	20 ms	262 µs	3662 µs	18,312%
0x015	20 ms	262 µs	3924 µs	19,62%
0x016	50 ms	262 µs	4186 µs	8,371%
0x017	50 ms	262 µs	4447 µs	8,894%
0x018	50 ms	262 µs	4709 µs	9,418%
0x019	50 ms	262 µs	4970 µs	9,941%
0x020	50 ms	262 µs	5232 µs	10,464%



115

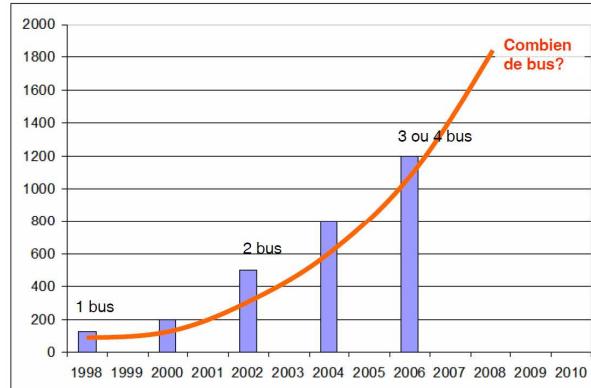
Limites du CAN (3/5) palliatifs

La solution la plus rapide à mettre en œuvre est d'ajouter des réseaux supplémentaires avec des passerelles ex:



116

Limites du CAN (4/5) progression du volume de données



NB: Ces chiffres ne sont spécifiques à aucun constructeur,
il s'agit d'une estimation moyenne

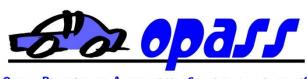


117

Limites du CAN (5/5) conclusion

La technologie CAN montre ses limites pour les questions suivantes:

- ⇒ La Charge élevée des différents bus empêche de garantir les temps de latence.
- ⇒ Il n'y a plus de marge de progression pour ajouter de nouvelles données.
- ⇒ L'apparition de GateWay a ajouté des délais supplémentaires entre les calculateurs.
- ⇒ Le niveau de complexité avec plusieurs réseaux devient difficile à appréhender. → Conception/Validation
- ⇒ Enfin le CAN n'est pas déterministe.
 - ⇒ Ne permet pas de réaliser de véritables fonctions X-By-Wire



118

BUS FLEXRAY



119

INTRODUCTION

- L'analyse des réseaux existants menée notamment par **BMW** et **Daimler Chrysler** à la fin des années 90 a montré que:
 - **CAN** ne possède pas un débit assez élevé pour les nouvelles applications et n'est ni déterministe ni redondant
 - **TTCAN** est déterministe mais manque de souplesse.
- **Le consortium FlexRay:**
-> Créé en **2000** par **BMW**, **DaimlerChrysler** et les fondateurs **Philips** et **Motorola Semiconductor (Freescale)**, **Core Members**, **Premium Members**.



120

FlexRay aujourd’hui

- Une spécification finale est figée V2.1
- Les composants (Philips, AMS) permettant de gérer la couche physique sont disponibles.
- Les micro-contrôleurs (Freescale; Philips; Renesas, Nec; TI; Infineon; Fujitsu...)permettant de gérer la couche protocole sont disponibles.
- Les outils de diagnostics Flexray existent aussi



121

FlexRay aujourd’hui

- BMW l'a intégré pour:
 - la suspension pilotée du X5 en 2007
 - Idem sur la nouvelle série 5 et série 7 en 2009
- Audi A8, Bentley Mulsanne, Rolls-Royce Ghost.
- General Motors est en train d'effectuer des tests sur des prototypes
- Daimler est en train d'étudier son intégration sur véhicule



122

Caractéristiques générales de FlexRay

FlexRay

- Pas de notion maître/esclave (multi maître)
- Débit élevé (max à 10 Mbits/s)
- Communication de type Time-Triggered (déterministe)
- Redondance de canaux
- Couche physique : Paire torsadée cuivre non blindée

Rappel sur le bus CAN

- Pas de notion maître/esclave (multi maître)
- Débit max à 1 Mbits/s
- Communication de type Event-Triggered
- Pas de redondance de canaux
- Couche physique : Paire torsadée cuivre non blindée



123

Rappel technique

Event-Triggered



- Dates d'émissions individuels
- Dates du départ et temps de latence inconnues à l'avance
- Nécessite un service d'arbitrage
- **Ex: CAN**

Time-Triggered



- Dates d'émission périodique
- Dates du départ et de latence connue à l'avance
- Nécessite un service de synchronisation
- **Ex: FlexRay**



124

Qu'est-ce que le déterminisme?

- *Un système est déterministe quand le comportement des sorties de celui-ci est parfaitement maîtrisé et ce quelles que soient ses entrées, on peut distinguer :*

=> Le déterminisme temporel lorsqu'il y a respect du timing,

=> Le déterminisme évènementiel lorsque tous les évènements sont traités.

Il découle de cette notion plusieurs autres :

=> La prévisibilité montre les possibilités que l'on a de prévoir comment le système va se comporter quelles que soient les circonstances.

=> L'urgence : il s'instaure une hiérarchie entre les différents traitements à effectuer ; certains étant plus importants que d'autres.



125

Pourquoi a-t-on besoin de déterminisme?

- **Synchronisation de données:**

Synchronisation de capteurs pour des applications de « perception » de l'environnement.

- **Maitrise des temps de latences:**

Augmentation du volume de données partagés sans risque d'augmentation des latences.

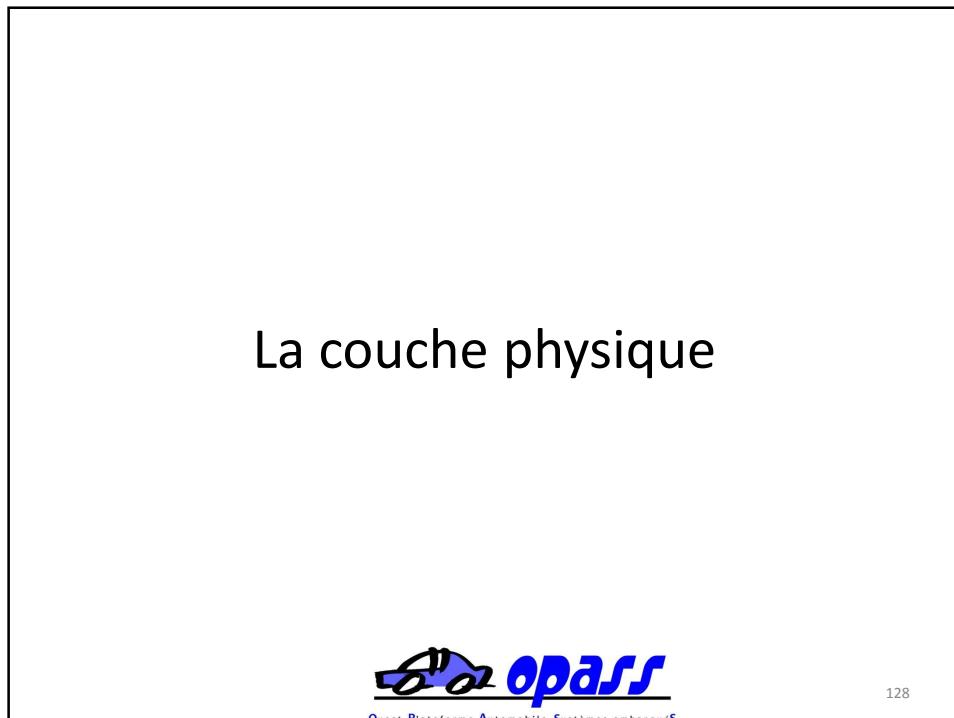
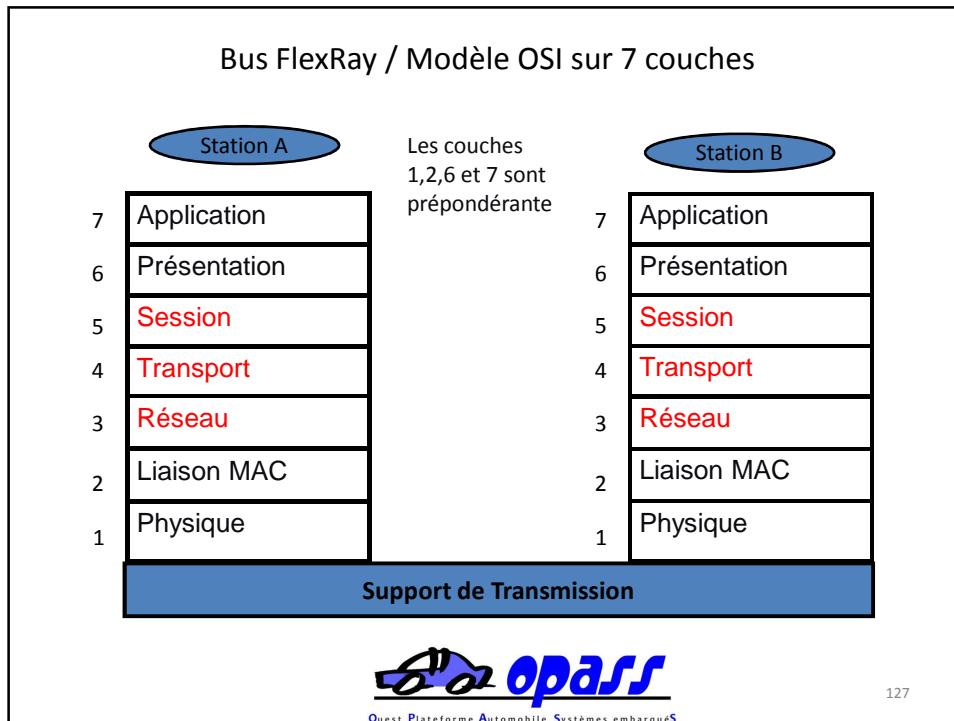
- **Diagnostique Rapide:**

Détection de perte de communication rapide afin d'appliquer les modes dégradés associés.

- En CAN il faut prendre une marge par rapport à la période de la trame pour la déclarer absente
- En FlexRay c'est immédiat



126



Accès au BUS

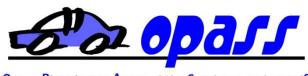
- Accès au BUS par multiplexage temporel via la Méthode TDMA(Time Division Multiple Access)
- Emission d'un mot de synchronisation par le nœud maître
- Emission des données par tous les nœuds à un intervalle de Temps précis
- Taille des données fixe



129

Temps de propagation

- Quelque soit la topologie:
 - Temps de Propagation Maximum : 2 500 ns
 - Utiliser des câbles d'impédance caractéristique comprise entre 80 et 110 ohms
 - Temps de propagation maximum: 10 ns/m
 - Atténuation max de 82 dB/km
 - Contribueurs à prendre en compte (connecteur, interface de ligne... étoiles actives...)

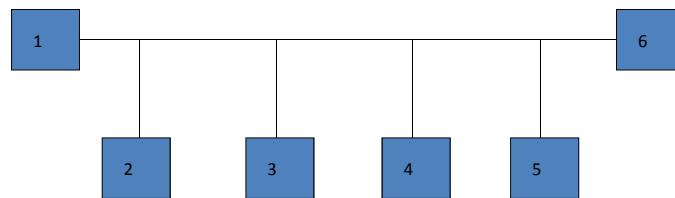


130

Topologie (1/2)

Topologie de type « **Etoile passive** (passive star) »:

- Habituellement appelée « épissure »
- Couramment utilisée sur le CAN

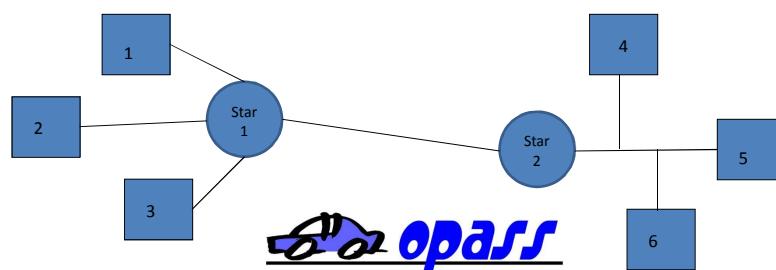


131

Topologie (2/2)

Topologie de type « **Etoile active** (active star) »:

- Appelée aussi « **Hub** » ou « **Répéteur** »
- Réalise une **remise en forme de signaux**: permet d'augmenter le nombre de noeuds et les longueurs de faisceau
- **Isole** les branches entre elles contre les courts-circuits
- Possibilité de faire du Diagnostic sur les branches étoiles
- Pour des raisons de coûts il est conseillé d'intégrer l'étoile active dans un noeud du réseau.



132

Exemple de connexions

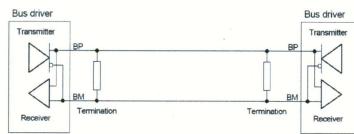


Fig. 1 Point to Point

- Transmission par signal différentiel
- Driver du bus = Emetteur et Récepteur
- Point à point
- Bus passif
- Etoile active et formes mixtes

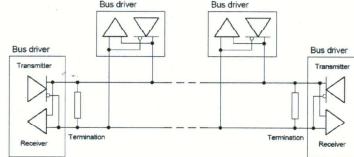


Fig. 2 Passive bus

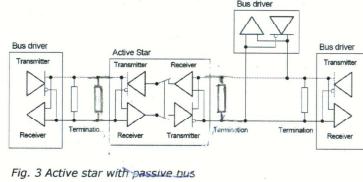


Fig. 3 Active star with passive bus

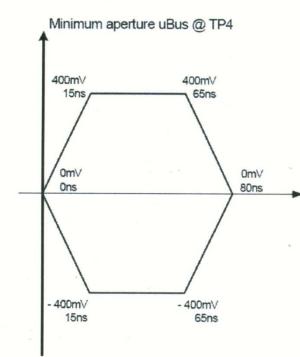
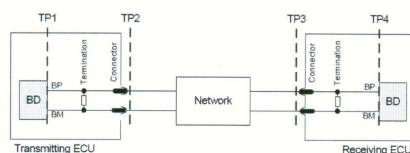
Source: FlexRay EPL2.1



133

Diagramme de l'œil (récepteur)

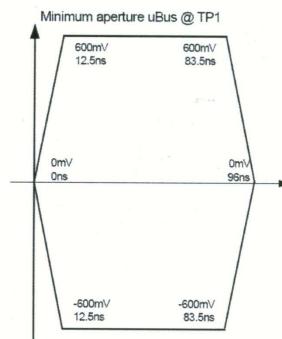
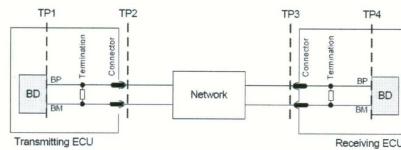
- Diagramme de l'œil du récepteur
- Valeur de tension selon les spécifications:
 - $|\text{u}_{\text{Bus,min}}| = 400 \text{ mV}$
 - $|\text{u}_{\text{Bus,max}}| = 2000 \text{ mV}$
- Représente « l'ouverture minimum de l'œil » du récepteur



134

Diagramme de l'œil (émetteur)

- ❑ Diagramme de l'œil de l'émetteur
- ❑ Valeur de tension selon les spécifications:
 - ❑ $|\text{uBus,min}| = 600 \text{ mV}$
 - ❑ $|\text{uBus,max}| = 2000 \text{ mV}$
- ❑ Représente « l'ouverture minimum de l'œil » de l'émetteur



135

LE PROTOCOLE



136

Composition des cycles

Composé de deux parties (segments temporels):

⇒ **Segment statique**

- ⇒ Émission déterministe
- ⇒ Communication en mode **Time Triggered**

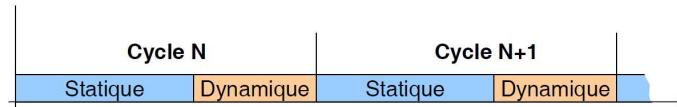


⇒ **Segment dynamique**

- ⇒ Émission non déterministe
- ⇒ Communication en mode **Event Triggered**
- ⇒ **Proche** du mode d'émission CAN



Les cycles s'enchaînent à période fixe pendant toute la durée du cycle applicatif

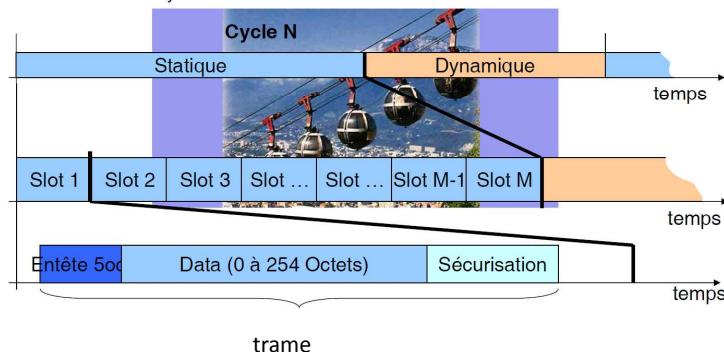


137

Le segment statique

Le Segment Statique:

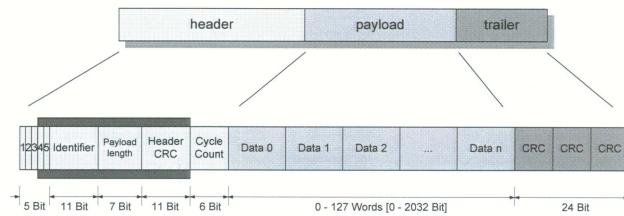
- ⇒ Découpé en tranches de temps fixe appelé "Slot"
- ⇒ Dans chaque slot une trame d'identifiant donné est émise
- ⇒ **Possibilité de charger le réseau à 100% sans augmenter la latence**
- ⇒ Taille de cycle et de slot fixe



138

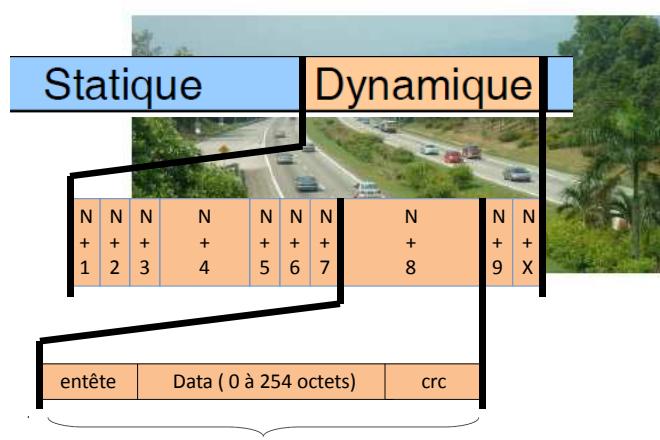
Le Format des Trames

- Deux CRC sur la trame
- Taille de la trame : 254 octets de données max. (+ 8 d'encapsulation)
- Les champs Data0 et Data1 peuvent servir au multiplexage des trames



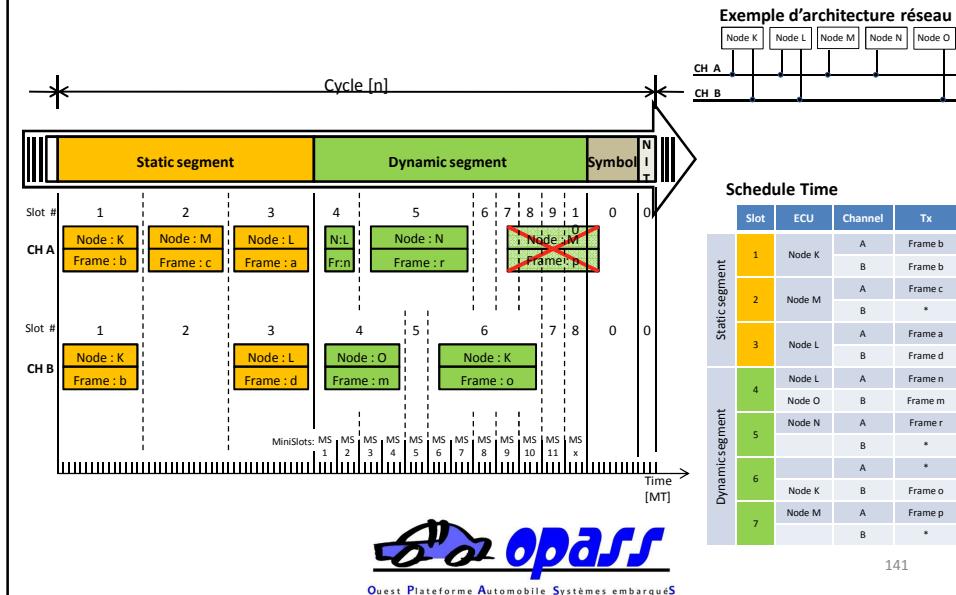
139

Le segment dynamique



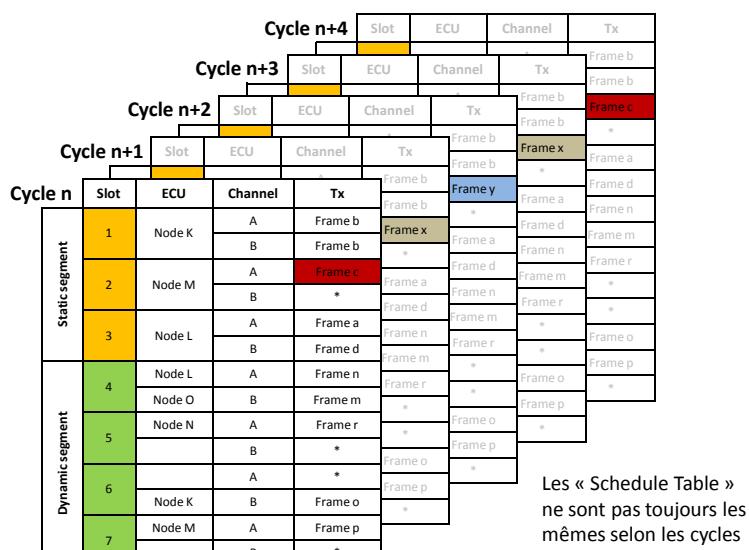
140

Le segment dynamique

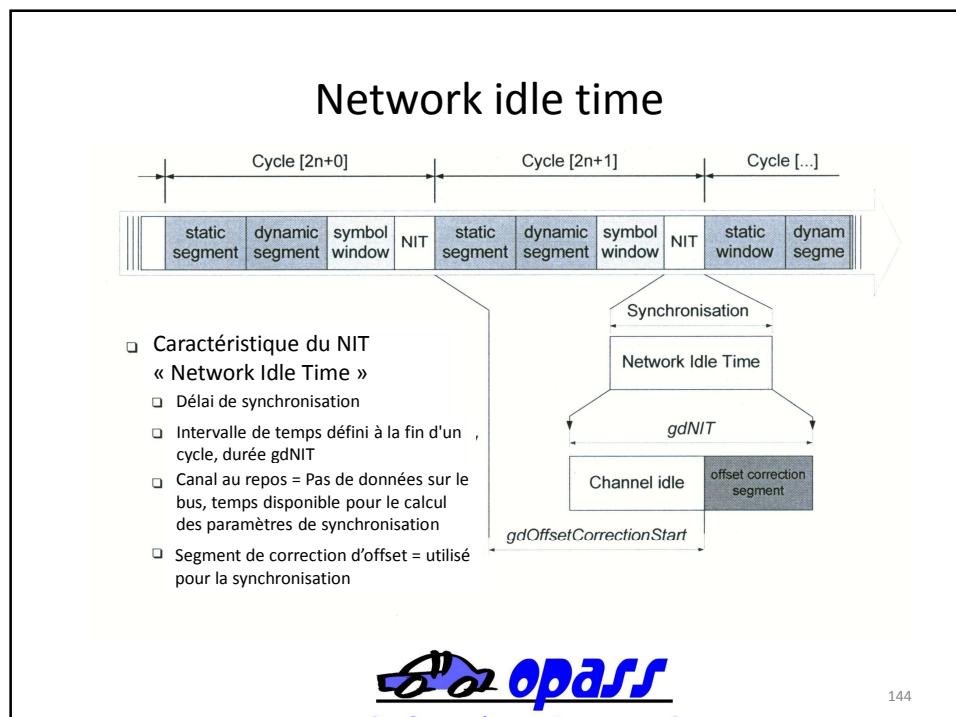
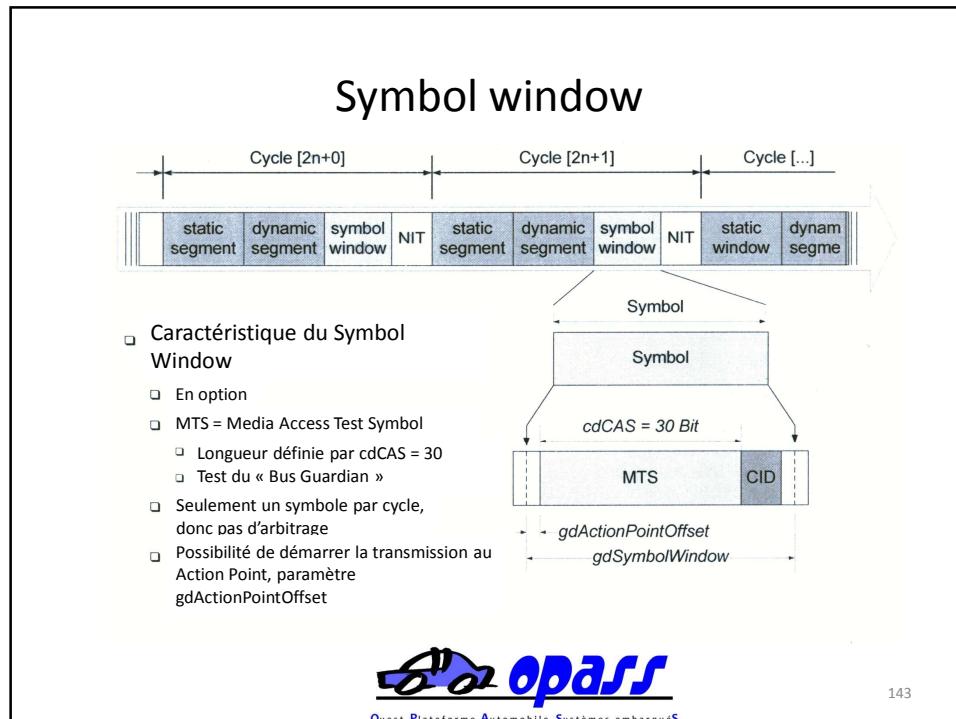


Ouest Plateforme Automobile Systèmes embarqués

Cycles



Ouest Plateforme Automobile Systèmes embarqués



Le Protocole Veille / Réveil : Principe

- **Veille**

=> Phase avant contact(+APC) tant que le bus n'est pas alimenté

- **Réveil**

=>Le démarrage complet du réseau s'effectue en deux étapes:

- > **Wake-Up**

- Un événement local provoque le réveil d'un noeud
- Un « wake-up symbol » provoque le réveil du réseau

- > **StartUp**

- Initialisation de la «Schedule Table»
- Le Noeud ColdStart initie la synchronisation
- Synchronisation du cycle de communication



145

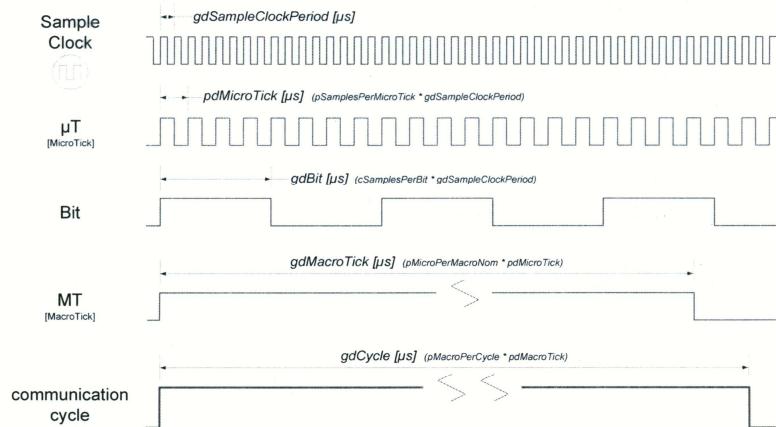
Le Protocole Start-up

- Commence lorsque tous les nœuds sont réveillés
- Le but est de démarrer et synchroniser le cycle de communication
- Deux types de nœuds :
 - Nœuds Coldstart
 - Nœuds non-Coldstart
- Deux Phases :
 - Startup des nœuds Coldstart
 - Un nœud Coldstart prend la main pour le démarrage
 - Les autres coldstart se synchronisent
 - Intégration des nœuds non-Coldstart



146

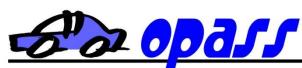
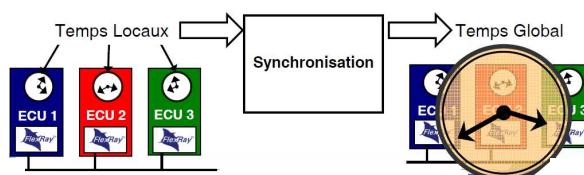
Le Protocole timing



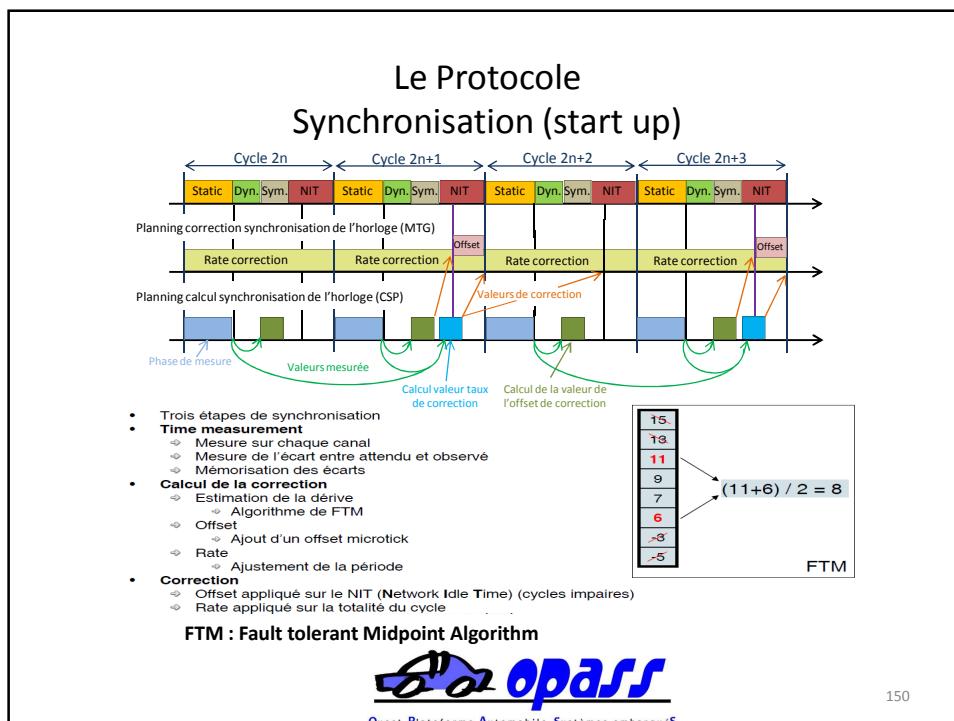
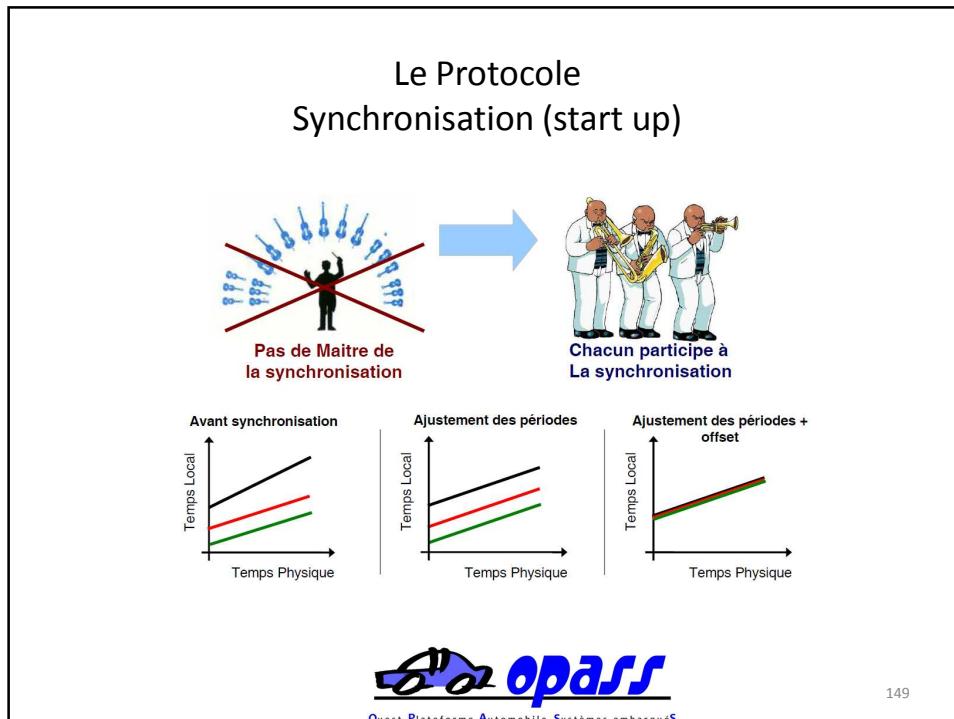
147

Le Protocole Synchronisation (start up)

- La **synchronisation** est un pré requis au fonctionnement d'un système **time-triggered**.
- Les variations de température de tension et la dispersion des caractéristiques des composants provoquent des **dérives d'horloges**.
- FlexRay utilise donc un mécanisme **d'horloge distribuée** sur le réseau pour tout les calculateurs=**Aucun n'est maître du temps**
- Chaque ECU mesure les temps des nœuds de synchronisation et en déduit une **base de temps Globale**. (Moyenne des horloges des nœuds de synchronisation)



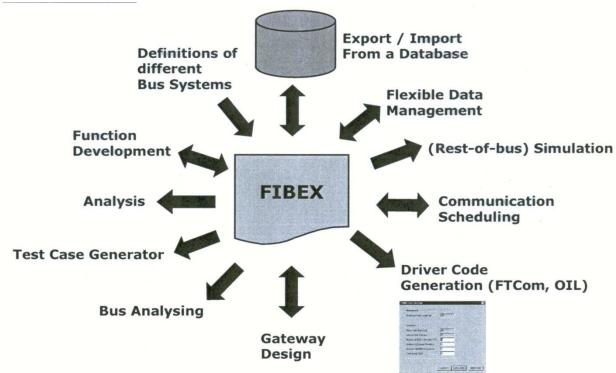
148



Fichier Système FlexRay Fibex (Field bus exchange format)

Bus supportés :

- Can
- Lin
- Flexray
- MOST

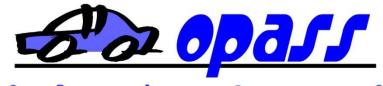


151

Démonstration bus Flexray



152



Ouest Plateforme Automobile Systèmes embarqués

Pour plus d'information :

Yannick Guyomarch

Tel : 06 76 84 48 54

Email : y.guyomarch @meito.com