

Les réseaux VAN - CAN





Sommaire

1. Introduction.....	4
1.1. Problématique	4
1.2. Solutions apportées par le multiplexage	5
2. Principes du multiplexage.....	7
2.1. Pyramide du CIM (Computer Integrated Manufacturing).....	7
2.2. Contraintes des bus terrains	8
2.3. Principe de l'échange de données.....	8
2.4. Structure d'une trame	9
2.5. Arbitrage d'une trame	10
2.6. CSMA / CA.....	11
2.7. Transmission sur un bus terrain	12
3. Les réseaux VAN	13
3.1. Topologie.....	13
3.2. Structure d'une trame	13
3.3. Services	15
3.4. Acteurs des échanges.....	18
3.5. Gestion des erreurs.....	19
3.6. Les modes VEILLE et REVEIL	20
4. Les réseaux CAN	21
4.1. Topologie.....	21
4.2. Structure d'une trame	22
4.3. Services	23
4.4. Gestion des erreurs.....	24
5. Bilan.....	27
5.1. Tableau Comparatif VAN – CAN.....	Erreur ! Signet non défini.
6. Webographie	28



Table des illustrations

Figure 1 : Exemple de câblage classique dans une voiture	5
Figure 2 : Exemple avec multiplexage	5
Figure 3 : Pyramide du CIM	7
Figure 4 : Schéma de l'émission d'une trame	8
Figure 5 : Arbitrage de trame	10
Figure 6 : Transmission classique	12
Figure 7 : Perturbation classique	12
Figure 8 : Transmission différentielle	12

1. Introduction

1.1. *Problématique*

L'électronique automobile est en évolution constante. En effet, de plus en plus de composants électroniques sont développés pour répondre aux exigences de plus en plus sévères en matière de pollution. De nouveaux équipements apparaissent également pour améliorer la sécurité et le confort du conducteur.

D'où une croissance constante ces 5 dernières années des fonctions d'électroniques :

- ABS, REF, MSR, ESP, ASR
- Direction à assistance variable, BVA, suspension pilotée, gestion moteur
- Airbag, anti-démarrage, clim. régulée, détection du sous gonflage des roues, aide au stationnement
- Allumage automatique des feux de croisement, essuie vitre automatique, correction de site des feux (lampes au Xénon)
- Allumage automatique des feux de détresse en cas de forte décélération ou de choc (1ère mondiale sur la Peugeot 607)
- Régulation de vitesse avec radar anti-collision, navigation par satellite
- Et à venir : direction et freins entièrement électrique, guidage du véhicule par rapport aux « bandes blanches », ...

Ce renforcement de l'électronique se traduit par :

- Une augmentation du nombre de calculateurs.
- Une augmentation du nombre de capteurs
- Une augmentation du nombre de faisceaux de câbles électriques.

1.2. Solutions apportées par le multiplexage

1.2.1. Réduction des coûts

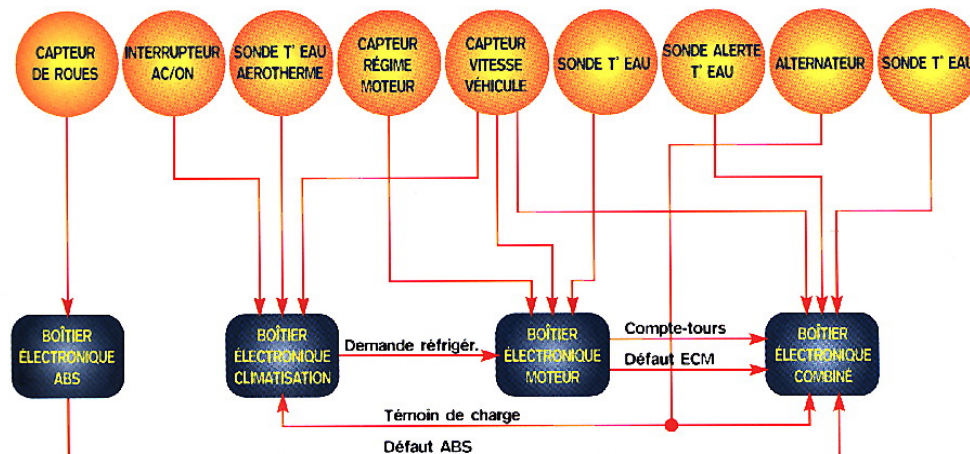


Figure 1 : Exemple de câblage classique dans une voiture

Certains capteurs ont des liaisons avec plusieurs calculateurs ou existent en deux exemplaires en raison de leur localisation. Les liaisons entre les boîtiers sont de plus en plus nombreuses.

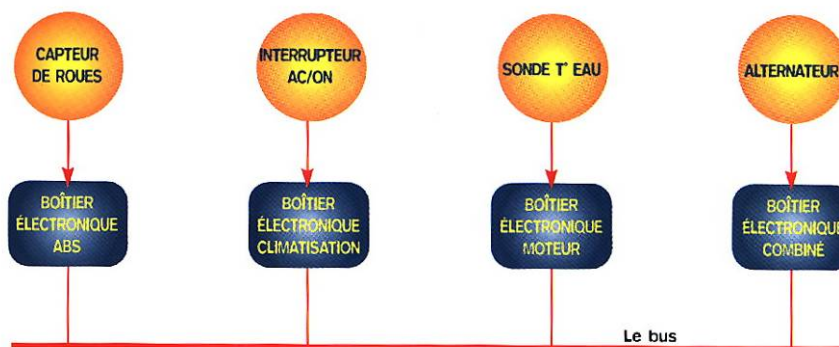


Figure 2 : Exemple avec multiplexage

Un partage d'information unique permet une synchronisation constante de tous les éléments du véhicule, mais surtout une diminution du nombre de capteurs et du câblage nécessaire pour les faire communiquer.



1.2.2. Augmentation de la qualité

La mise réseau et la réduction des coûts de câblage permettent une augmentation de la qualité des matériaux utilisés, mais offrent aussi la possibilité de mettre en place des outils de diagnostic centralisé.

La qualité des équipements est donc globalement améliorée pour un coût de production équivalent.

1.2.3. Meilleure évolutivité

La gestion centralisée de l'information permet un ajout simplifié de nouveaux composants et permet ainsi une évolution plus rapide des options des véhicules.

1.2.4. Normalisation des communications

La normalisation induite par le partage du média d'information par plusieurs composants permet aux constructeurs, qui ne sont en réalité que des « assembleurs », de simplifier l'architecture des véhicules.

2. Principes du multiplexage

Les bus de terrain VAN et CAN définissent les couches Physique et Liaison du modèle OSI. Au dessus de ces couches, différents protocoles ont été développés pour répondre aux besoins particuliers :

- KWP2000 : Diagnostique de panne (VAN / Ligne K)
- Diagnostic on CAN : Diagnostique de panne (CAN)
- CANOpen / DeviceNet : Application de CAN dans l'industrie

2.1. Pyramide du CIM (Computer Integrated Manufacturing)

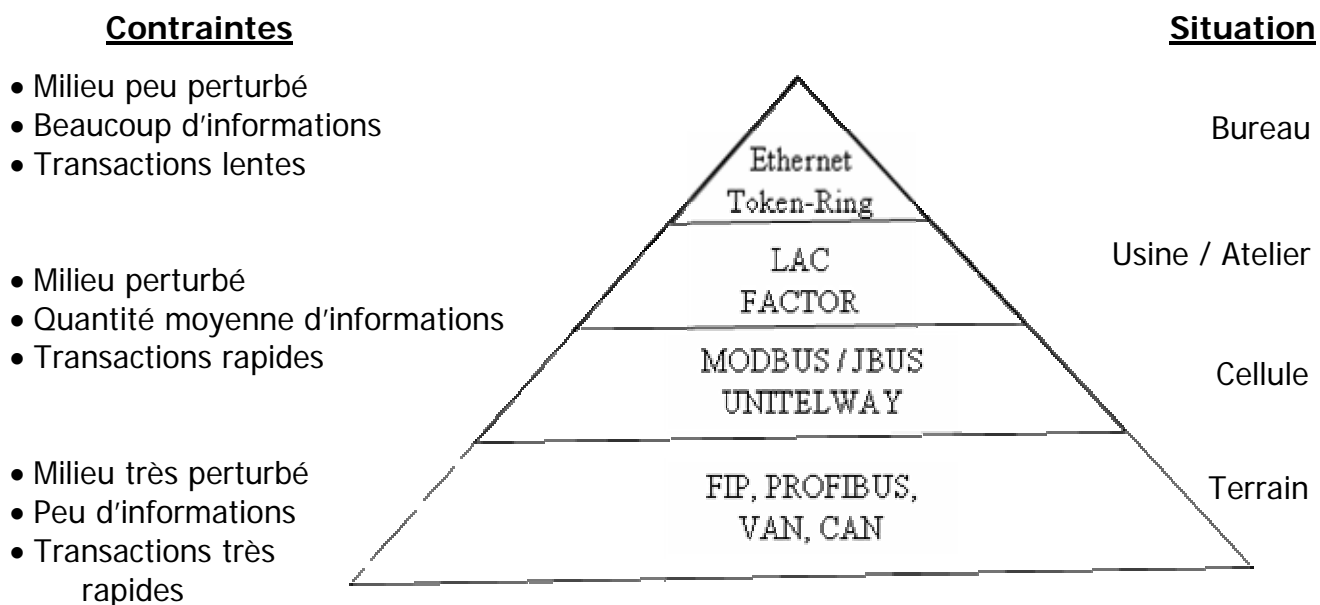


Figure 3 : Pyramide du CIM

2.2. Contraintes des bus terrains

Les bus terrains sont soumis à des contraintes difficiles.

Leur environnement tout d'abord est très perturbé. En effet, les véhicules sont très chargés en perturbations électromagnétiques, mais engendrent également des vibrations et de variations de températures sont importantes.

Par ailleurs, les enjeux en terme de fiabilité sont très importants étant donné les besoins existant au niveau sécurité et satisfaction client.

Néanmoins, peu d'informations sont à faire circuler puisque les fréquences d'échanges sont assez longues. Les transactions sont également très rapides puisqu'elles ne représentent que des échanges commande – contrôle.

2.3. Principe de l'échange de données

Les BUS terrain sont basés sur un mécanisme de communication en broadcast. Il n'existe pas de notion de machine ni d'adresse, mais simplement de contenu d'information. Chaque message échangé comporte un identifiant unique à tout le réseau qui identifie l'information transmise ainsi que son niveau de priorité. Cette notion est très importante quand plusieurs éléments se partagent l'accès au média.

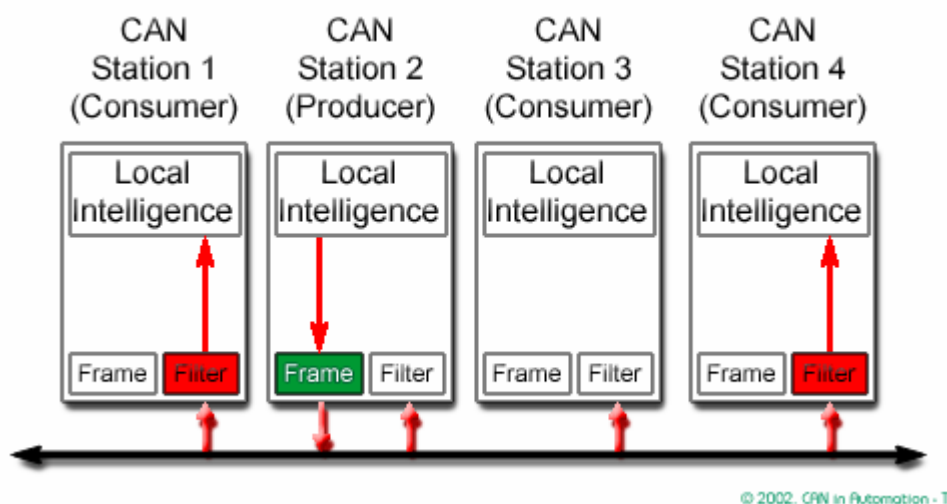


Figure 4 : Schéma de l'émission d'une trame

2.4. Structure d'une trame

Début	ID	COM	Données	Contrôle	Fin data	Ack	Fin
-------	----	-----	---------	----------	----------	-----	-----

Le champ **début** est le symbole indiquant le début d'une trame. C'est ce champ qui permettra également de synchroniser les horloges.

Le champ **identificateur** sert à identifier le contenu du message. Ce n'est en rien une adresse puisqu'un même composant pourra recevoir des messages avec des identifiants différents en fonction de son masque de sélection. De même, un élément pourra émettre des trames avec des identifiants différents en fonction de l'information qu'il veut transmettre.

Ce champ permet en réalité de connaître l'information que contient le message. Comme chaque information est plus ou moins importante, c'est sur ce champ que sera déterminée la priorité du message, comme nous le verrons par la suite.

Le champ **COM** est le champ de commande qui annonce la nature du message. Il permet de définir quel type de message est transporté.

Le champ **données** contient les données à transmettre. Ce champ peut faire jusqu'à 28 octets en VAN.

Le champ **contrôle** vérifie la cohérence de la trame. C'est un checksum de 15 bits sur les données de la trame.

Le champ **Fin data** permet de déterminer où les données s'arrêtent dans la trame. Il consiste en une violation du code Manchester. Ce champ n'existe qu'en VAN.

Le champ **ACK** est un accusé de réception si aucune erreur n'a été détectée.

Le champ **Fin** contient un symbole indiquant la fin de la trame.

2.5. Arbitrage d'une trame

Dans le traitement en temps réel, l'urgence des messages à échanger au-dessus du réseau peut différer considérablement : une dimension changeante rapidement, le régime moteur par exemple, doit être transmise plus fréquemment et donc avec moins de délai que d'autres dimensions comme la température d'eau.

Cette priorité est définie par l'identifiant du message. Ces priorités sont définies durant la conception d'un système par l'intermédiaire de séquences binaires et ne peuvent être changées dynamiquement. L'élément ayant la valeur binaire la plus petite a la priorité la plus élevée.

Les conflits d'accès au média sont résolus par un arbitrage bit à bit de la part de chaque station à l'écoute du média. Ceci peut être réalisé grâce à mécanisme électrique par lequel les états dominants écrasent les états récessifs.

En effet, tout nœud émettant un bit récessif et recevant simultanément un bit dominant perd la parole. Il devient alors récepteur d'un message ayant une priorité supérieure et ne reparlera pas tant que le média n'est pas disponible.

Les demandes de transmission sont manipulées par ordre de leur importance pour le système dans l'ensemble. Ceci se prouve particulièrement avantageux dans des situations de surcharge. Puisque l'accès au média est accordé par priorité sur la base des messages, il est possible de garantir des temps de latence faible dans les systèmes en temps réel.

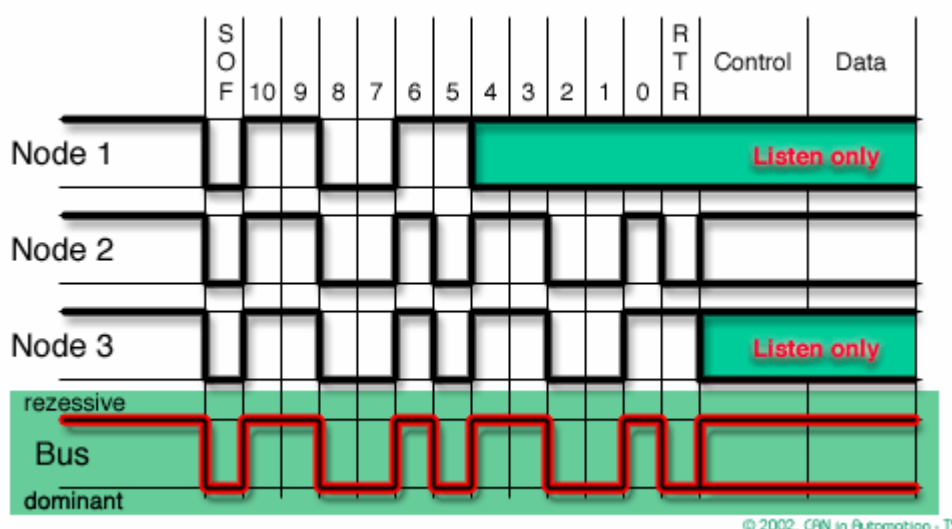


Figure 5 : Arbitrage de trame

2.6. CSMA / CA

La sensibilité des données pouvant circuler sur le média, il n'est pas envisageable d'avoir des pertes de données. Pour éviter cela, les bus terrain implémentent CSMA/CA. Mais que signifie ce sigle ?

CSMA/CA = Carrier Sense Medium Access with Collision Avoidance

Cela signifie tout d'abord qu'un nœud ne prendra pas la parole si un autre est déjà en train de transmettre une information. Pour pouvoir émettre un message, un élément devra donc tout d'abord écouter le média. Si personne ne parle, il commencera à émettre. Si quelqu'un occupe déjà le média, le nœud attendra un temps aléatoire avant de tenter à nouveau d'émettre. Ce fonctionnement est semblable à Ethernet qui fait aussi du CSMA.

Néanmoins, il peut arriver que deux machines débutent une transmission exactement au même instant. C'est là que les bus terrain vont avoir un fonctionnement différent d'Ethernet. En effet, en Ethernet, la collision due à cette émission simultanée sera détectée et la trame sera réémise plus tard (Collision Detect). Dans le cas des bus terrain, la collision sera esquivée et le nœud le plus prioritaire gardera la parole (Collision Avoidance).

Pour que cela soit possible, il faut que la longueur du bus, *cad* la distance entre les deux nœuds les plus éloignés, ne dépasse pas une valeur maximale qui dépend du bit. La durée de transmission maximale ne peut pas excéder entre le début du bit et le point d'échantillonnage du niveau sur le bus. Les retards éventuels induits par la traversée des composants doivent être prise en compte.

Pour rappel et pour comparaison, la longueur maximale d'un réseau Ethernet (CSMA/CD) est déterminée par la taille de la trame la plus courte qui peut circuler et du temps qu'il faut pour la transmettre. En CSMA/CA, elle est donnée en fonction du bit.

2.7. Transmission sur un bus terrain

Etant donné les perturbations possibles sur un bus terrain, il a fallu mettre en place une protection supplémentaire au blindage des câbles ou à l'utilisation de câbles torsadés. Pour cela, c'est la façon dont le signal est transmis qui a été modifié.

En effet, en temps normal, un signal électrique est transmis à l'aide de deux fils :

- l'un est la masse de référence
- l'autre porte le signal utile

Malheureusement, si une interférence intervient sur la ligne au cours du transport, elle n'est pas détectable et peut entraîner une déformation du signal.



Figure 6 : Transmission classique



Figure 7 : Perturbation classique

Sur les bus terrains utilisés par VAN et CAN, les deux fils portent des signaux utiles. L'un est le signal réel, l'autre, ce même signal inversé. A la réception, il suffit de soustraire les deux messages pour retrouver le signal d'origine.

Ainsi, si une perturbation a modifié le signal, elle aura modifié les deux signaux et va donc se soustraire elle-même à la réception.

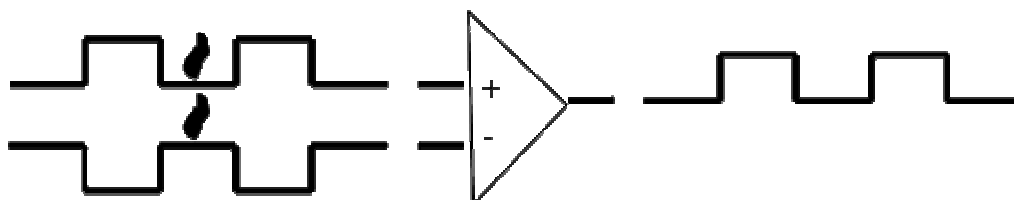
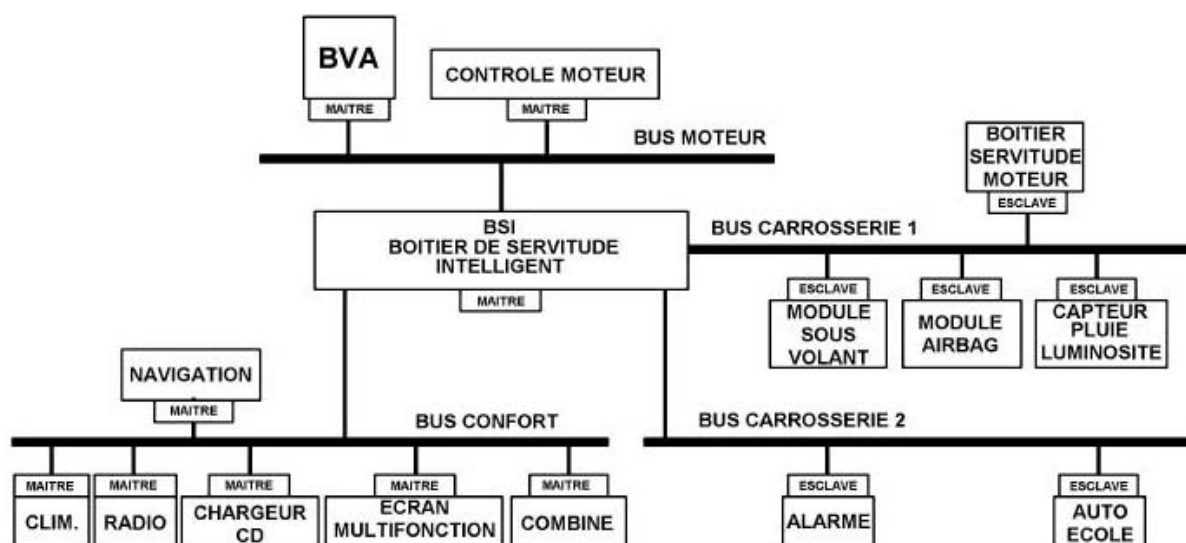


Figure 8 : Transmission différentielle

3. Les réseaux VAN



3.1. Topologie

La topologie des réseaux VAN est dite « Arbre – Bus ». C'est la mieux adaptée aux contraintes automobiles. Le bus VAN n'impose pas de restriction sur la taille des branches et n'impose pas de terminaison de ligne. Seule la distance entre les deux éléments les plus éloignés est limitée par le débit du bus.

Les topologies étoile et anneau sont possibles avec le VAN mais ne sont pas utilisées dans l'automobile. Elles imposent des restrictions qui ne sont pas compatibles dans l'automobile.

3.2. Structure d'une trame

La structure d'une trame VAN est semblable à celle présentée dans le chapitre 2.4. Les champs propres à l'implémentation VAN sont les champs COM et FIN DATA.

3.2.1. Champ COM

Le champ COM permet de spécifier le type de Services désiré. Les services offerts par VAN améliorent la rapidité des transferts d'informations. De plus, ils permettent de répondre plus précisément aux demandes des éléments maîtres.

Voici le détail de ce champ :

Les services	Le champ COM			
	EXT	RAK	R/W	RTR
1- trame de données sans acquittement	R	D	D	D
1- trame de données avec acquittement	R	R	D	D
1- trame de demande différée	R	R	R	R
1- trame de réponse différée	R	R	R	D
1- réponse dans la trame	R	R	R	D

3.2.2. Champ FIN DATA

Le champ FIN DATA permet de définir là où les données s'arrêtent dans la trame. Dans l'implémentation VAN, il consiste à effectuer un viol du codage Manchester. Il est utilisé par l'acteur Producteur pour spécifier au Consommateur qu'il a fini de remplir les données.

3.2.3. Taille des champs

Le champ Début dure 10 Time Slot (4 TS dominant, 4 TS récessif, 1 TS dominant et 1 TS récessif), c'est ce qui permet aux autres nœuds du réseau de se synchroniser. La durée d'un TS est égale à 1/débit. De plus, le codage utilisé est le codage Manchester, donc tous les 4TS, on rajoute un bit. L'identificateur de la trame est codé sur 12 bits. Le champ COM qui permet de définir la nature du service utilisé est

composé de 4 bits. Le champ Data peut contenir jusqu'à 28 octets et le calcul du CRC se fait sur 15 bits. La violation du codage Manchester sur le champ Contrôle se fait sur 2 bits et le champ Fin de Trame contient 8 bits.

Tableau récapitulatif :

Début : 10 bits (4D-4R-1D-1R)

Identificateur : 12 bits

Com : 4 bits

Data : 0 à 28 octets

Contrôle : 15 bits

Fin data : 2 bits

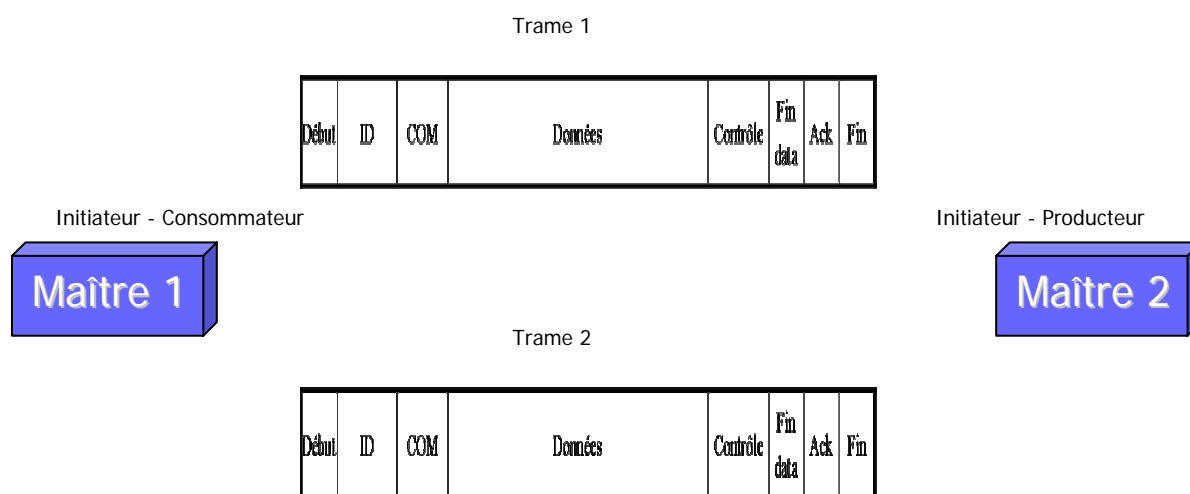
Fin de trame : 8 bits

Espace minimal inter trame : 8TS

3.3. Services

3.3.1. Trame de données sans acquittement

Aussi appelé trame de diffusion, le contenu du message est susceptible de concerner plusieurs machines.



3.3.2. Trame de données avec acquittement

Le destinataire du message est tenu de répondre à l'expéditeur. Si des erreurs sont détectées dans la trame, l'esclave ne renvoi pas d'acquiescement. On parle aussi de transmission Point à Point.

3.3.3. Demande de transmission distante

Réponse dans la trame

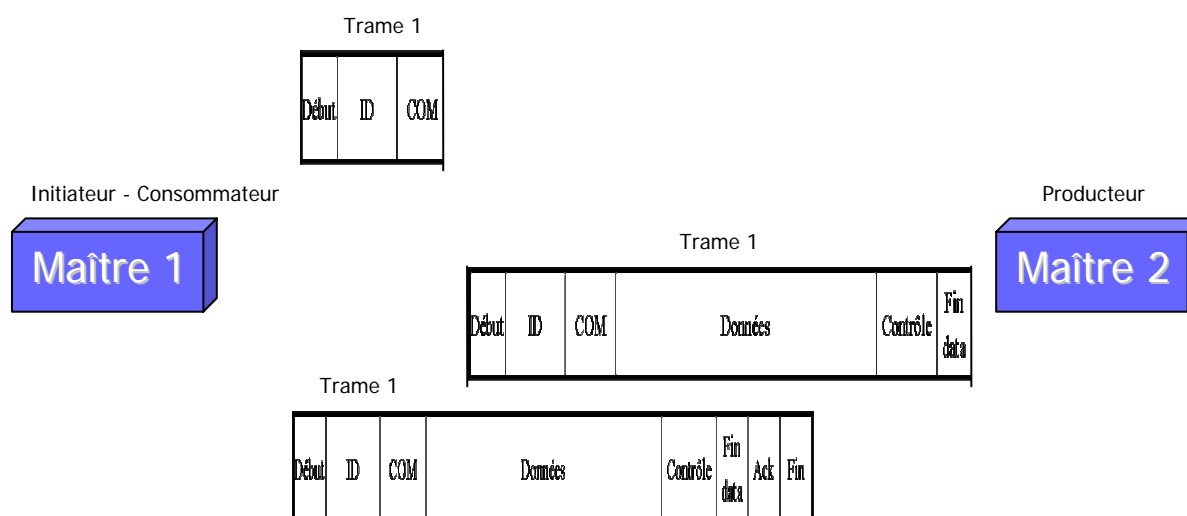
La réponse dans la trame est un service particulier du VAN qui permet un échange très rapide des données. Voici le scénario représentant l'utilisation de ce service.

Le maître place sur le bus le début de la trame destinée au régime moteur.

Les esclaves analysent les champs identificateur et Com de la trame et seul l'esclave en question continue de lire la trame.

L'esclave place ensuite les données demandées par le maître dans la trame suivi du champ CRC et EOD.

Lorsque le maître détecte le champ EOD, il acquitte ensuite la trame puis place le champ EOF.

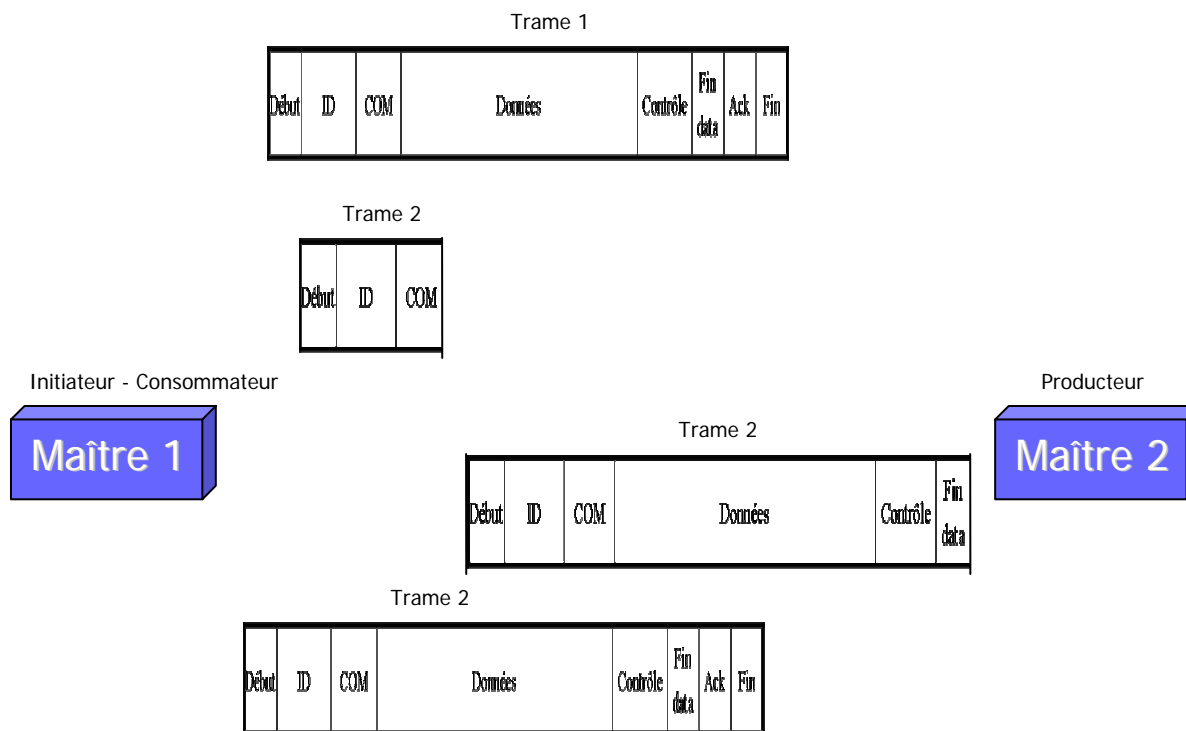




Dans ce cas, une seule trame est émise sur le bus, elle contient à la fois la demande et la réponse. Ce mécanisme améliore considérablement la rapidité des échanges. Le maître 2 est alors que Producteur, il place seulement les données, les champs contrôle et FIN DATA sur le bus.

Demande différée

Le principe de la demande différée est similaire à la réponse dans la trame mis a part que la demande est effectuée plus tôt pour prévenir l'élément esclave qu'il souhaite des informations. Le schéma ci-dessous explique ce principe :



3.4. Acteurs des échanges

En VAN, 3 acteurs interviennent lors de l'échange d'informations sur le bus. Le premier acteur qui intervient lors d'un échange est **l'Initiateur**. Il commence à effectuer une demande de synchronisation des horloges à tous les éléments du réseau (Champ Début). Il place ensuite le service demandé puis la nature du service. C'est ensuite à l'acteur **Producteur** de travailler. Il va placer les données dans la champ DATA puis dans le champ FIN DATA (viol du code Manchester). L'acteur **Consommateur** pourra ensuite lire le message et placer les champs ACK (si nécessaire) et le champ FIN TRAME.

3.5. Gestion des erreurs

Voici les différents **types d'erreurs détectées** par les réseaux VAN :

- ERREUR BIT : bit émis différent du bit reçu.

Cette erreur est une erreur que l'on peut considérer comme attendue. En effet, une erreur bit survient à chaque fois qu'il y a arbitrage sur le bus.

- ERREUR DE CODE : violation de code

L'erreur de code est également une erreur « attendue ». Cette erreur apparaît à chaque fois qu'un viol du codage Manchester est effectué. Par exemple, le champ FIN DATA effectue tout le temps une violation de codage pour signaler au consommateur des données qu'il a fini de remplir le champ.

- ERREUR DE FORMAT : Erreur de codage ou de synchronisation sur les champs fixes (SOF, EOF, ACK, EOD)

Cette erreur survient à chaque fois qu'il y a un problème dans la structure de la trame. Si celle-ci n'est pas conforme à la structure d'une trame VAN, une erreur est envoyée.

- ERREUR DE CRC : CRC reçu différent de celui calculé.

Cette erreur survient lorsque le consommateur calcule le CRC du message et que celui-ci est différent de celui placé par le producteur.

- ERREUR ACK : Champ ACK non conforme à la demande.

Cette erreur apparaît lorsque le message acquitté n'est pas celui attendu.

Attention, une erreur est détectée :

Lorsqu'une erreur est détectée, il se pose 2 alternatives, soit le nœud en question est en réception, auquel cas il ne prend pas en compte la trame, soit le nœud est en transmission, et alors il stoppe son émission, passe en réception jusqu'à la détection d'un IFS et réessaie de transmettre le message. (Le nombre de répétition de la trame peut varier entre 0 et 7. Il faut le programmer au niveau du gestionnaire de protocole).



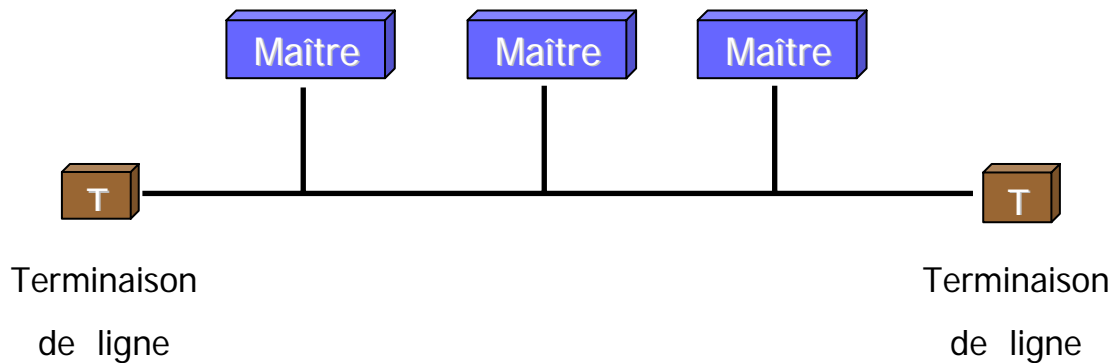
3.6. Les modes VEILLE et REVEIL

Le mode VEILLE du réseau VAN permet de limiter la consommation de l'électronique du véhicule au maximum pendant les phases d'arrêt tout en permettant une remise en service du réseau demandé par un nœud non alimenté.

Tous les modules du réseau VAN sont alimentés par une tension +VAN qui est pilotée par un module maître. La coupure de cette alimentation provoque la mise en veille des esclaves. Une ligne spéciale est utilisée pour transmettre la demande de réveil d'un esclave par un appel de courant.

4. Les réseaux CAN

4.1. Topologie



Contrairement au réseau VAN, le réseau CAN ne supporte qu'une seule topologie, il ne fonctionne que sur un bus série auquel il faut donc ajouter des terminaison de ligne.

De type multi-maître uniquement, il n'utilise pas la notion d'esclaves. Chaque composant est capable d'émettre, de recevoir et de traiter les messages.

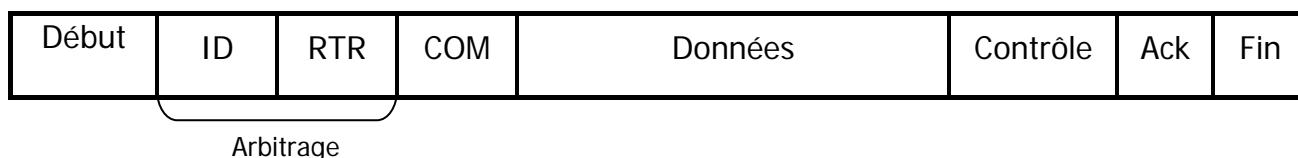
La longueur maximum du bus est déterminée par la charge capacitive et le débit. Les configurations recommandées sont les suivantes sachant que la norme spécifie un débit maximum de 1Mbit/s:

Débit	Longueur
1Mbit/s	40m
500Kbit/s	100m
100Kbit/s	500m
20Kbit/s	1 000m

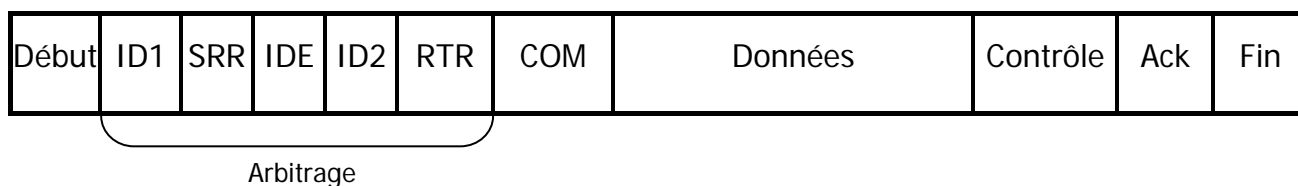
4.2. Structure d'une trame

La structure d'une trame CAN est semblable à celle présentée dans le chapitre 2.4. Cependant il faut remarquer que le protocole CAN 2.0 comporte deux spécifications qui diffèrent uniquement au niveau de l'ID. La version 2.0A définit des ID de 11 bits et il est question alors de trames standards (« Standard Frames ») et la version 2.0B des ID de 29 bits pour former dans ce cas là des trames étendues (« Extended Frames »).

2.0A



2.0B



Nous allons reprendre ci-dessous quelques champs qui ont une particularité pour CAN.

4.2.1. Le Champ d'arbitrage

Le champ arbitrage est constitué de l'identificateur et du bit RTR.

L'ID permet d'identifier le message, il fait 11 bits et est transmis de ID10 à ID0 (ID0 étant le bit le moins significatif).

Le RTR (Remote Transmission Request) est utilisé pour les messages de demandes de transmission (voir 4.3.2).



4.2.2. Le Champ Com

Le champ COM est composé de 6 bits. Les 2 premiers sont réservés et les 4 suivants vont indiquer le nombre d'octets du champ de données. Ces 4 bits sont appelés le DLC (Data Length Code) et il va donc forcément être compris entre 0 et 8.

4.2.3. Le Champ Ack

Le champ ACK est composé de 2 bits, le ACK Slot et le ACK Delimiter (1 bit récessif). Un nœud en train de transmettre place un bit récessif dans le ACK Slot. Un nœud qui reçoit correctement le message envoie un bit dominant pendant le ACK Slot.

4.3. Services

En CAN il n'existe que 2 Types de services et ceux-ci dépendent du bit RTR. Il existe en plus deux autres types de messages, les messages d'erreurs et les messages de surcharge.

4.3.1. Transmission de données

La transmission de données permet d'envoyer des informations sur le bus, à l'aide d'une trame de données. Le bit RTR est dominant

4.3.2. Demande de transmission

Les demandes de transmission distantes permettent de faire une demande d'informations en précisant un identificateur, à l'aide d'une trame de requête. Le bit RTR est récessif.

La réponse à cette trame de demande est une trame de donnée, où le même identificateur est utilisé. Le bit RTR est dominant.

4.3.3. Les autres types de messages

- Les messages d'erreurs sont transmis par un nœud qui a détecté une erreur. Leur fonctionnement est détaillé ci-après (voir 4.4).
- Les messages de surcharge (Overload frames) permettent de créer un délai entre deux trames de données ou deux trames de demandes successives.

4.4. Gestion des erreurs

L'environnement du CAN est difficile, aussi il a été créé avec des mécanismes de détection d'erreurs efficaces.

Les différents types d'erreur sont les mêmes que dans VAN aussi vous pouvez les retrouver au chapitre 3.5.

Nous allons nous intéresser plus en détail à la trame d'erreur, mais également à toute la gestion qui est mise en place dans ce genre de cas.

4.4.1. Trame d'erreur

Il existe pour une trame d'erreur 2 parties bien distinctes.

- La première partie appelée « Error Flag » est construite par tous les nœuds. Elle consiste dans l'envoi d'une trame de 6 bits dominant consécutif, or le protocole CAN fonctionnant avec le principe du code NRZ (pas plus de 5 bits dominants consécutifs) détecte une violation de ce code. Les autres nœuds détectent alors une erreur et envoient à leur tour un « Error Flag ». Il y a alors sur le bus une superposition de ces messages d'erreurs. La séquence de bits dominants varie entre 6 et 12 bits.

Il existe de plus 2 types d'Error flags :

- Active error flag : 6 bits dominants



- Passive error flag : 6 bits récessifs, écrasés ensuite par des bits dominants.

- La seconde partie est appelée « Error Delimiter », elle est construite avec 8 bits récessifs. Lorsque chaque nœud a terminé d'émettre son « error flag », il émet des bits récessifs qui sont écrasés tant que d'autres nœuds émettent des bits dominants. Les nœuds écoutent le bus jusqu'à ce qu'il détectent un bit récessif, il envoient alors 7 bits récessifs supplémentaires.

4.4.2. Les Compteurs d'erreurs

Deux compteurs d'erreurs sont implémentés dans chaque nœud d'un réseau CAN. Celui des erreurs de transmission et celui des erreurs de réception.

Le fait d'enregistrer le nombre d'erreurs générées ou détectées par un nœud va permettre d'établir une différence entre les erreurs dites temporaires et les erreurs dites permanentes qui peuvent être dues à un mauvais fonctionnement matériel.

Il va donc être possible de retirer le nœud défaillant du réseau.

L'incrémementation et la décrémementation de ces compteurs suivent des règles précises :

- Lorsqu'un nœud est en train d'écouter le réseau (mode récepteur) et qu'il détecte une erreur, alors son compteur d'erreurs de réception est incrémenté de 1.

- Lorsqu'un nœud est en train d'écrire sur le réseau et qu'il envoie un « Error Flag » alors son compteur d'erreurs de transmission est incrémenté de 8.

- Après chaque transmission réussie, le compteur d'erreurs de transmission est décrémenté de 1.

- Après chaque réception réussie, le compteur d'erreurs de réception est décrémenté de 1.

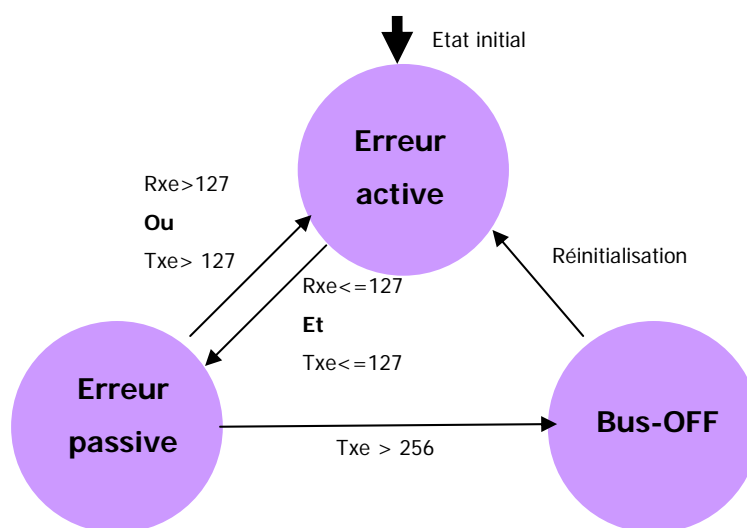
4.4.3. Les différents états d'envoi d'erreurs

Les valeurs des compteurs d'erreurs vont être utilisées ensuite pour connaître l'état dans lequel se trouve un nœud pour la transmission d'erreur.

Ainsi on obtient ceci :

- Entre 0 et 127 pour les 2 compteurs : Le nœud est dans un **état actif**, il participe normalement à la communication sur le bus. Il émettra un « Active Error Flag » s'il détecte une erreur.
- Entre 128 et de 256 pour l'un des 2 compteurs : Le nœud est dans un **état passif**, il participe normalement à la communication sur le bus mais s'il détecte une erreur, il émettra un « Passive Error Flag » qui ne sera pas pris en compte par les autres nœud et qui sera dans le cas d'une véritable erreur recouvert par une « Active Error Flag ».
- Au delà de 256 pour le compteur de transmission : Le nœud passe dans l'état **buss-off**, il est alors déconnecté du réseau et ne pourra plus participer à la communication sur le bus. Il faudra le réinitialiser manuellement pour qu'il puisse reprendre sa place sur le réseau.

Résumé :



5. Bilan – Tableau comparatif VAN - CAN

	VAN	CAN
Topologie	libre	fixe
Services	variés	limités
Gestion des erreurs	limité	efficace
Codage	Manchester	NRZ
Gestion de veille	oui	aucune

En VAN, il est possible de créer une topologie de réseau en Arbre, en Bus, en étoile ou en anneau, il n'y a aucune restriction. Cela permet de réaliser plusieurs BUS distinct en séparant les fonctionnalités (bus confort, bus sécurité, etc...). En CAN, on ne peut créer qu'un seul bus série. De plus, on ne peut pas mettre en place des noeuds esclaves, tous les organes du réseau sont des éléments maitres.

Concernant les services, en VAN plusieurs types de services sont proposés, notamment le service de réponse dans la trame qui améliore considérablement la durée des échanges d'informations.

Les types d'erreurs détectés en VAN et en CAN sont identiques, néanmoins, la gestion des erreurs est beaucoup plus efficace en CAN car si un élément est défectueux, le mécanisme d'incrémentation des compteurs d'erreurs va rendre inactif cet élément.



6. Webographie

http://www.geea.org/article.php3?id_article=183&men=rub

<http://pboursin.club.fr/pdgmpx.htm>

<http://www.can-cia.org>

<http://www.educauto.org/>

<http://www.oberle.org/can-index.html>

<http://edelaunay.chez.tiscali.fr/buscan.htm>