

# Pilote de bateau TP32 / Bus CAN

## ANNEXE 2 : DESCRIPTION de la communication par BUS CAN

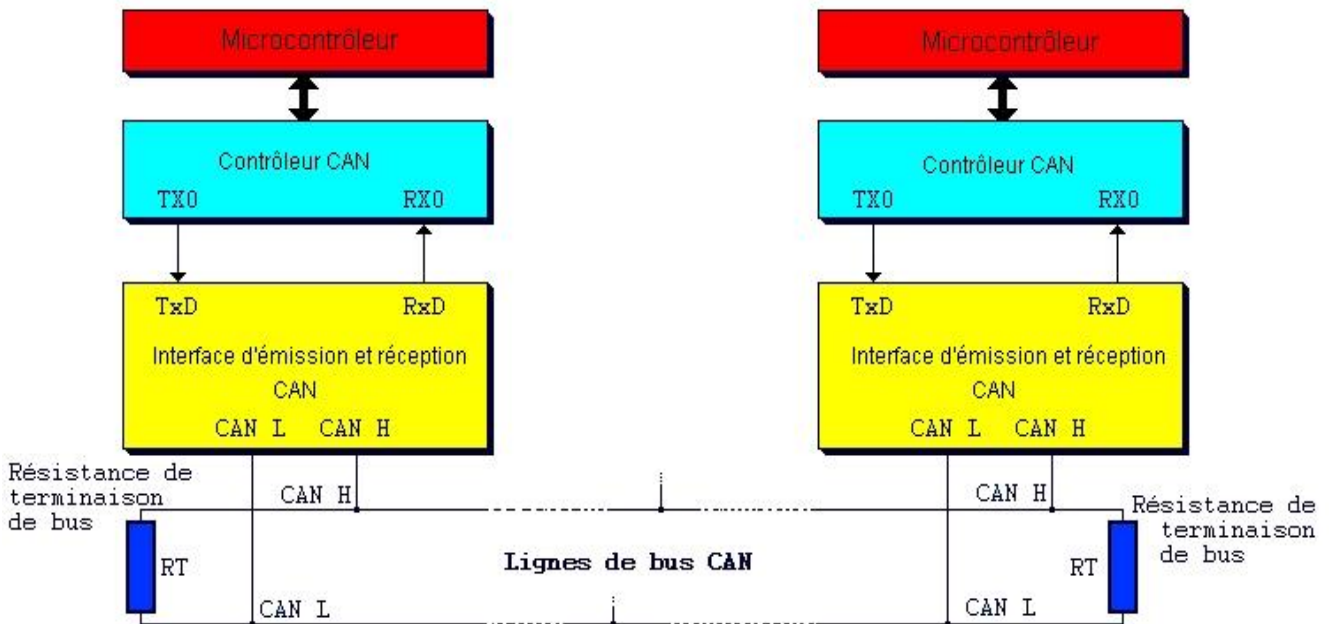
### Origine et intérêt du bus CAN

Le bus CAN (Controller Area Network) a été développé par l'industrie automobile pour faire communiquer de manière série les différents systèmes électroniques embarqués. Auparavant, tous les organes de commandes des véhicules échangeaient les données par l'intermédiaire de lignes dédiées, mais l'augmentation du nombre d'organes embarqués a contraint les équipementiers automobiles à développer une nouvelle architecture de communication en réseau.

PSA avec des partenaires comme Sagem et Valeo ont développé le bus VAN (Vehicule Area Network). En Allemagne, Bosch a développé, au milieu des années 80, le bus CAN ou "Controller Area Network".

Avec le bus CAN, les différents organes (organes de commande, capteurs ou actionneurs) sont reliés par un bus. série. Le protocole CAN de base leur permet d'échanger 2048 variables.

La transmission physique peut s'effectuer sur une paire filaire torsadée ou par liaison infrarouge, hertzienne ou par fibre optique.



Le protocole utilisé permet de détecter et corriger des erreurs induites sur le câble par des radiations électromagnétiques (CEM) dues par exemple au fonctionnement des moteurs électriques embarqués.

L'organisation en réseau apporte la possibilité de se connecter en un point (prise diagnostic) et de communiquer avec toute l'électronique de manière à détecter les pannes ou modifier une configuration.

### Principe de fonctionnement d'un bus CAN :

Les données doivent être transmises rapidement afin d'être traitées en quasi temps réel le débit de transmission peut atteindre 1 Mbit/s. L'assignation du bus en cas de conflits doit être rapide, lorsque plusieurs stations souhaitent transmettre simultanément des messages.

Le protocole est basé sur le principe de diffusion générale, lors de l'émission. Chaque message est reçu par tous les récepteurs mais comporte un identificateur.

Grâce à cet identificateur, les stations, en permanence à l'écoute, reconnaissent et traitent les messages qui les concernent; elles ignorent les autres.

L'urgence des informations échangées sur le bus peut être très diverse. Une valeur variant rapidement, comme l'état d'un capteur de sécurité (ABS) doit être transmise prioritairement et avec un retard moindre que la température du moteur, qui évolue lentement. Les priorités sont attribuées lors de l'analyse conceptuelle du réseau, au moyen de valeurs binaires.

Sur le réseau CAN, l'identificateur de chaque message, mot de 11 bits (format standard) ou de 29 bits (format étendu), détermine sa priorité.

On définit jusqu'à 2048 messages plus ou moins prioritaires sur le réseau avec les 11 bits d'identification (en format standard).

Chaque message peut contenir jusqu'à 8 octets de données, par exemple les états de 64 capteurs TOR.

# Pilote de bateau TP32 / Bus CAN

## Principe de l'arbitrage d'un bus CAN :

L'accès au bus CAN suit la technique CSMA/CD (écoute de chaque station avant de parler mais pas de tour de parole, résolution des collisions par priorité).

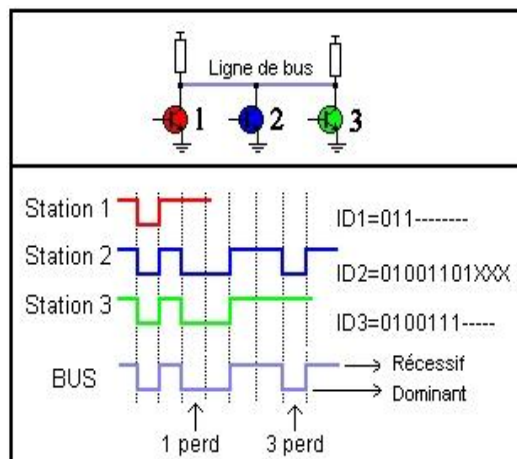
En cas d'émission simultanée de plusieurs stations, l'attribution du bus suit le principe d'arbitrage suivant.

Les stations émettant simultanément sur le bus, comparent bit à bit l'identificateur de leur message (ID) avec celui des messages concurrents.

Les ID de priorités moins élevée perdent la compétition face à celle qui a la priorité la plus élevée, pour cela les stations sont câblées sur le bus par le principe du "ET câblé". En cas de conflit, c'est à dire émission simultanée, la valeur 0 écrase la valeur 1.

On appelle donc "état dominant" l'état logique 0, et "état récessif" l'état logique 1. Lors de l'arbitrage bit à bit, dès qu'une station émettrice se trouve en état récessif et détecte un état dominant, « elle perd » et arrête d'émettre.

Tous les perdants deviennent automatiquement des récepteurs du message, et ne tentent à nouveau d'émettre que lorsque le bus se libère. Dans l'exemple ci-dessus, trois stations émettent simultanément, la station 1 perd la compétition puis la station 3. Seule la station 2 pourra transmettre.

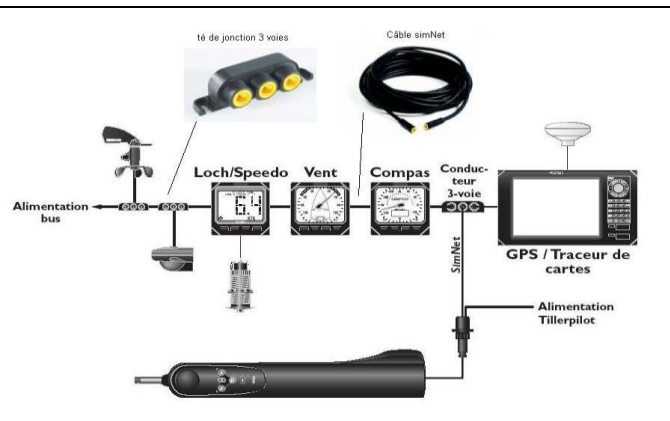


## Caractéristiques physiques du bus CAN :

### Installation SimNet standard

Tous les appareils SimNet (instruments, traceurs de cartes, radars, pilotes automatiques, etc.) utilisent le même câble unique de bus haute vitesse pour partager les données.

Le système est "plug-and-play" : aucun paramétrage n'est nécessaire, il suffit de brancher le pilote automatique au port SimNet libre le plus proche pour qu'il détecte automatiquement les données exploitables disponibles sur le réseau.



### Le câble utilisé dans le réseau SimNet

Le câble est constitué par deux paires de fils torsadés.

Les fils torsadés permettent de limiter les parasites auxquels un bus de terrain est généralement soumis.

Une première paire transporte l'alimentation électrique pour les appareils à faible consommation : **CAN 0V (masse) et CAN +V (+12V)**

La deuxième paire supporte les signaux de données : **CAN L (CAN LOW) et CAN H (CAN HIGH)**

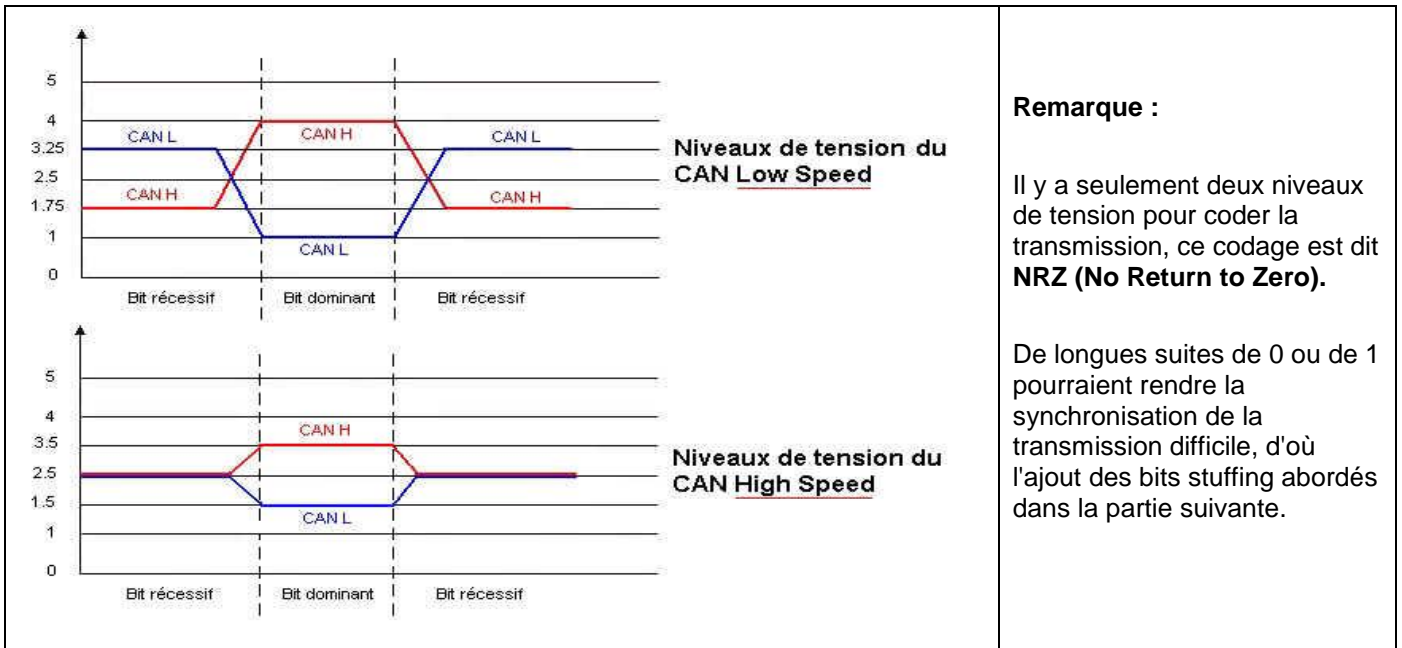


### Niveaux de tension et débit de transmission

Pour les niveaux physiques sur le bus, il est important de distinguer les deux types de transmission possibles :  Transmission en bus CAN <b>low speed</b>  Transmission en bus CAN <b>high speed</b> .	Paramètres	<b>CAN low speed</b>	<b>CAN high speed</b>
	Débit max	125 kb/s	1 Mb/s
	Niveau dominant	<b>CAN H = 4V</b> <b>CAN L = 1V</b>	<b>V<sub>CAN H</sub> - V<sub>CAN L</sub> = 2V</b>
	Niveau récessif	<b>CAN H = 1,75V</b> <b>CAN L = 3,25V</b>	<b>V<sub>CAN H</sub> - V<sub>CAN L</sub> = 0V</b>
	Tensions d'alimentation	5V	5V

Les chronogrammes ci-dessous détaillent les niveaux de tension selon l'état de la ligne et le type de bus CAN.

# Pilote de bateau TP32 / Bus CAN



## Formats de trames de messages :

La norme CAN définit deux formats de protocole :

- Standard (Version 2.0 A) : identificateur (ID) de 11 bits
- Étendu (Version 2.0 B) identificateur 29 bits qui permet l'augmentation du nombre de stations sur le réseau.

Le nombre d'octets de données échangés à chaque trame reste inchangé.

# Pilote de bateau TP32 / Bus CAN

## ANNEXE 3 : Trame du RESEAU SIMNET au FORMAT ETENDU

Le pilote envoie périodiquement différentes trames. L'une de ces trames contient l'information CAP.  
Elle sera exploitée par un instrument indicateur de cap du type SIMRAD IS20.



### Trame de données au format étendu

Start of Frame	Champ d'arbitrage	Champ de contrôle	Champ de données	Champ CRC	Champ d'acquiescement	End of Frame
1 bit dominant	29 bits + 3bits	6 bits	0 à 8 octets 8 octets sur SimNet	16 bits	2 bits	7 bits récessifs

Une trame est composée des champs suivants :

- Le champ de début de trame **SOF** (*Start Of Frame*), 1 bit dominant.
- Le champ d'arbitrage :

Poids forts de l'identificateur	SRR	IDE	Poids faibles de l'identificateur	RTR
11 bits	1 bit	1 bit	18 bits	1 bit

- SRR** (*Substitute Remote Request*),
- IDE** (*Identifier Extension bit*) établit la distinction entre format standard (état dominant) et format étendu (état récessif)
- RTR** (*Remote Transmission Request*) détermine s'il s'agit d'une trame de données ou d'une trame de demande de message.

- Le champ de contrôle :

R1	R0	DLC3	DLC2	DLC1	DLC0
1 bit dominant	1 bit dominant	1 bit	1 bit	1 bit	1 bit

- R1 et R0 sont des bits réservés,**
- Les quatre bits DLC (*Data Length Code*) permettent de coder le nombre d'octets contenus dans la zone de données.
- 0 correspond à DLC3=d; DLC2=d; DLC1=d; DLC0=d (d pour dominant; r pour récessif) et 8 correspond à DLC3=r; DLC2=d; DLC1=d; DLC0=d

- zone de vérification de la validité de la trame **CRC** (*Cyclic Redundancy Code*) :

Séquence de CRC	Délimiteur CRC
15 bits	1 bit récessif

Ces bits sont recalculés à la réception et comparés aux bits reçus ; S'il y a une différence, une erreur CRC est déclarée.

- zone d'acquiescement (**ACKnowledge**)

Bit d'acquiescement	Délimiteur d'acquiescement
1 bit	1 bit récessif

L'émetteur positionne sur la ligne un bit récessif. S'il a bien reçu cette trame, le récepteur adressé force alors ce bit à l'état dominant.

- zone de fin de trame **EOF** (*End Of Frame*), 7 bits récessifs.

Remarque : Lors de la construction d'une trame, si 5 bits sont consécutivement au même état (0 ou 1), un bit supplémentaire (**Stuff bit**) d'état complémentaire est inséré dans la trame.

Exemple: 000010011111011110