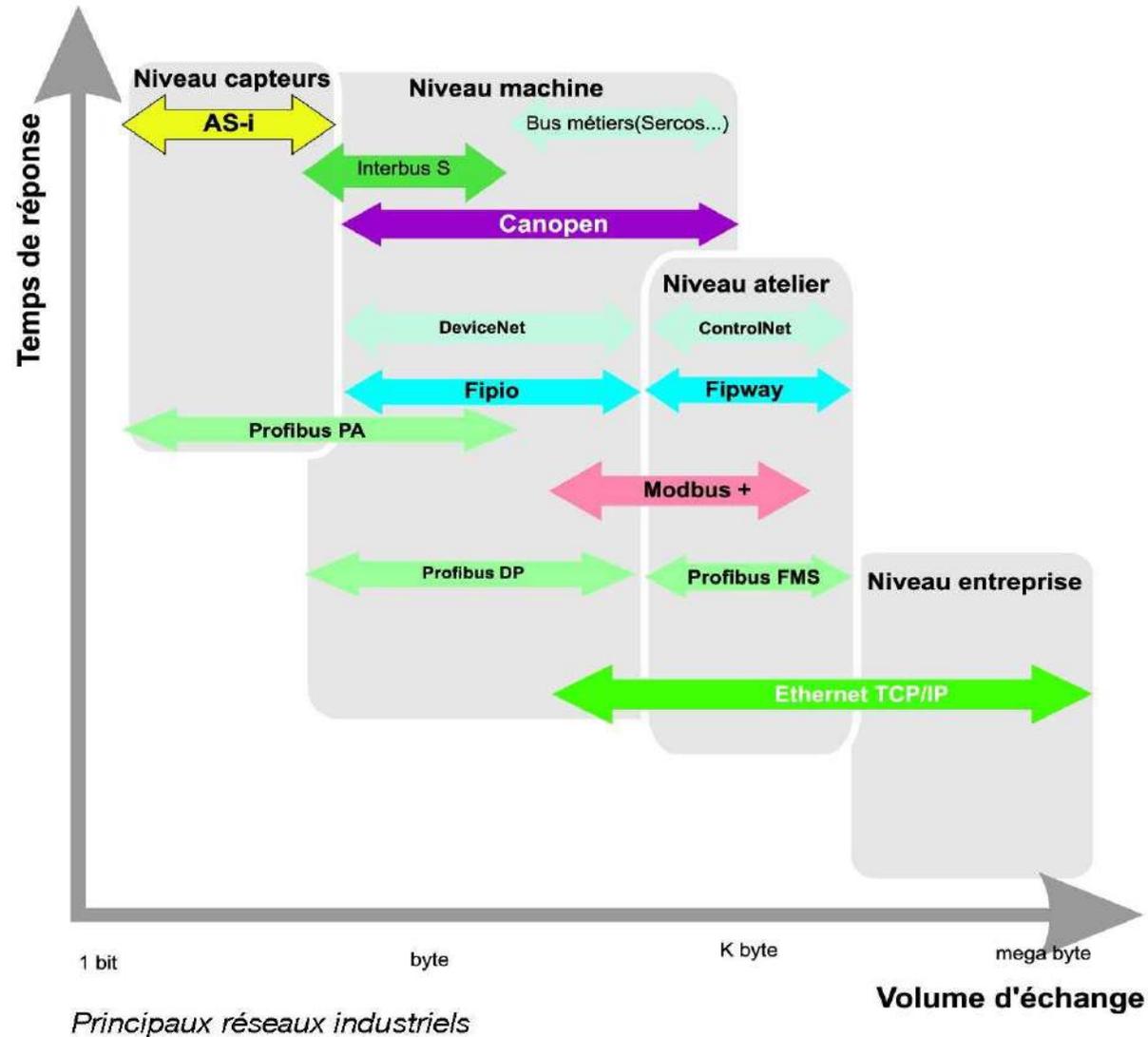


Réseaux de terrain : 3^o partie



La liaison RS-485 (1)

EIA-485 (souvent appelée RS-485) est une norme qui définit les caractéristiques électriques de la couche physique d'une interface numérique série.

Ce qu'il faut retenir : la liaison RS-485 a toutes les caractéristiques d'un bus informatique (liaison multi-point numérique série) permettant d'interconnecter plusieurs dispositifs (jusqu'à 32 émetteurs et 32 récepteurs) avec 2 fils ("half duplex") ou 4 fils ("full duplex") sur des distances maximales de l'ordre du kilomètre en mode différentiel (qui permet d'obtenir une meilleure tolérance aux perturbations) avec un débit élevé (jusqu'à 10Mbits/s).

De nombreuses interfaces d'interconnexion dans le monde informatique et industriel sont basées sur EIA-485 : Profibus, Modbus, etc ...



La liaison RS-485 (2)

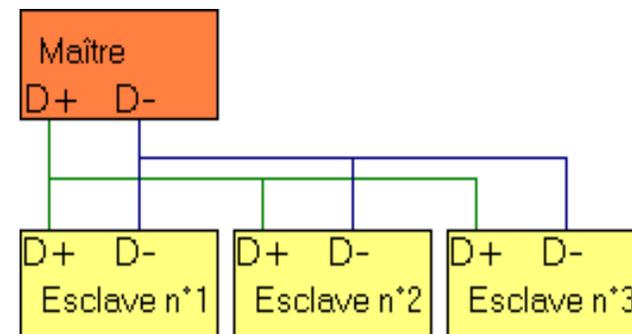
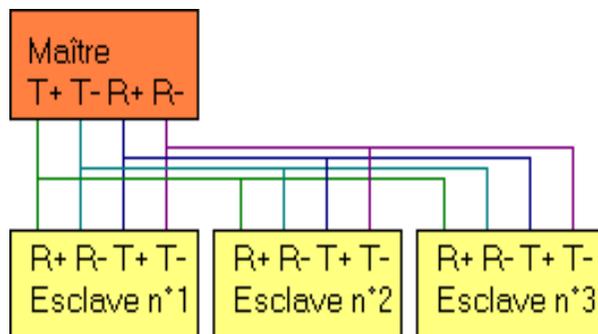
Liaisons	Topologie		Mode		Distance	
	Point à point	Multipoint	Différentiel	Unipolaire	< 100 m	> 1000 m
RS-232	x			x	x	
RS-422	X (1)	X (2)	x			x
RS-485		x	x			x

Remarques concernant la RS422 :

Les 2 réponses sont acceptées car :

- (1) Une liaison RS422 est essentiellement utilisée pour allonger une liaison RS232 et est considérée à ce titre comme une liaison Point à Point.
- (2) Une liaison RS422 est considérée comme une fausse liaison multipoint (simplex) : dans ce cas un seul équipement est connecté au bus et peut transmettre à tous les autres.

Exemple d'utilisation de la liaison RS-485 avec le protocole MODBUS :



Modbus (1)

Modbus est un protocole de communication utilisé pour des réseaux d'automates programmables (API).

Il fonctionne sur le mode maître / esclave.

Le protocole Modbus peut être implémenté :

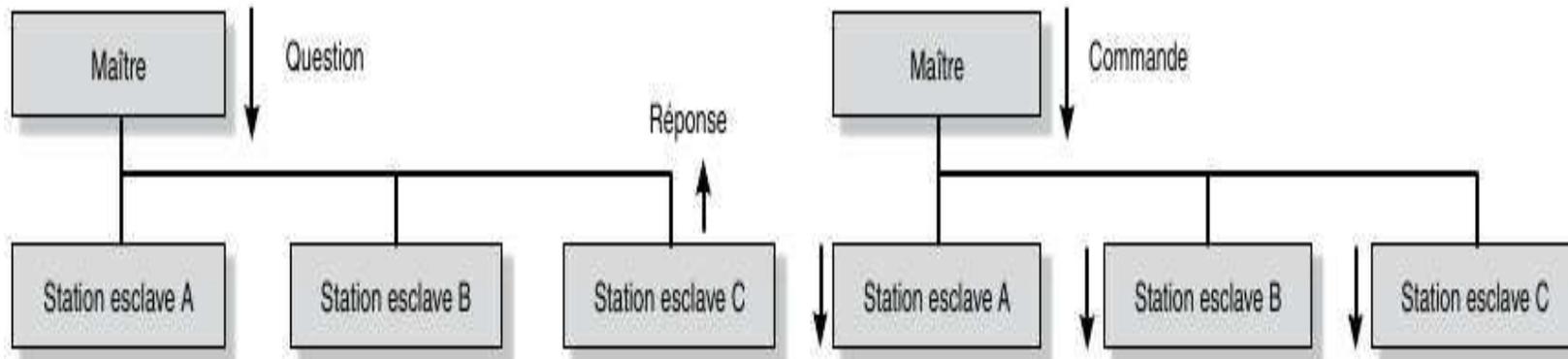
- sur une liaison série asynchrone de type RS-422 ou RS-485 ou TTY (boucle de courant), avec des débits et sur des distances variables ;*
- via TCP/IP sur Ethernet ; on parle alors de Modbus TCP/IP ;*
- via Modbus Plus. Modbus Plus est un réseau à passage de jetons à 1 Mb/s, pouvant transporter les trames Modbus et d'autres services propre à ce réseau.*

Usage courant : une liaison multipoints de type RS-485 relie maître et esclaves via une paire différentielle qui permet un débit élevé (jusqu'à 10 Mbits/s) sur une distance importante (jusqu'à 1 200 m). Elle ne dispose que de 2 bornes qui alternativement passent les données dans un sens puis dans l'autre (half duplex).



Modbus (2)

Le maître envoie une demande et attend une réponse.



Il ne peut y avoir sur la ligne qu'un seul équipement en train d'émettre.

Aucun esclave ne peut envoyer un message sans une demande préalable du maître.

Le dialogue entre les esclaves est impossible.

Le maître peut diffuser un message à tous les esclaves présents sur le réseau (diffusion générale ou broadcast). Pour cela, il utilise l'adresse 0.



Modbus (3)

Le maître envoie un message vers un esclave. Il répond par un message du même type. La détection de fin de message est réalisée sur un silence de plus de 2 caractères.

La question

Elle contient un code fonction indiquant à l'esclave adressé le type d'action demandé.

Les données contiennent des informations complémentaires dont l'esclave a besoin pour exécuter cette fonction.

Le mot de contrôle permet à l'esclave de s'assurer de l'intégralité du contenu de la question.

La réponse

Si une erreur apparaît, le code fonction est modifié pour indiquer que la réponse est une réponse d'exception (MSB*=0 : pas d'erreur ; MSB=1 : erreur).

Les données contiennent alors un code (code d'exception) permettant de connaître le type d'erreur.

Code d'exception :

- 01 Fonction illégale (erreur sur le code fonction)
- 02 Erreur sur l'adresse du registre ou du coil
- 08 Erreur de transmission (suite au contrôle du CRC ou du Timing)

Question :

N° station esclave	Code fonction + bit d'erreur	Information spécifique concernant la demande	Mot de contrôle
1 octet	1 octet	n octets	2 octets

Réponse :

N° station esclave	Code fonction + bit d'erreur	Données transmises	Mot de contrôle
1 octet	1 octet	n octets	2 octets

Réponse lors d'une erreur :

N° station esclave	Code fonction + bit d'erreur	Code d'exception	Mot de contrôle
1 octet	1 octet	1 octet	2 octets

*MSB : Most Significant Bit



Modbus (4)

MODBUS offre 19 fonctions différentes. Tous les équipements ne supportent pas tous les codes fonction.

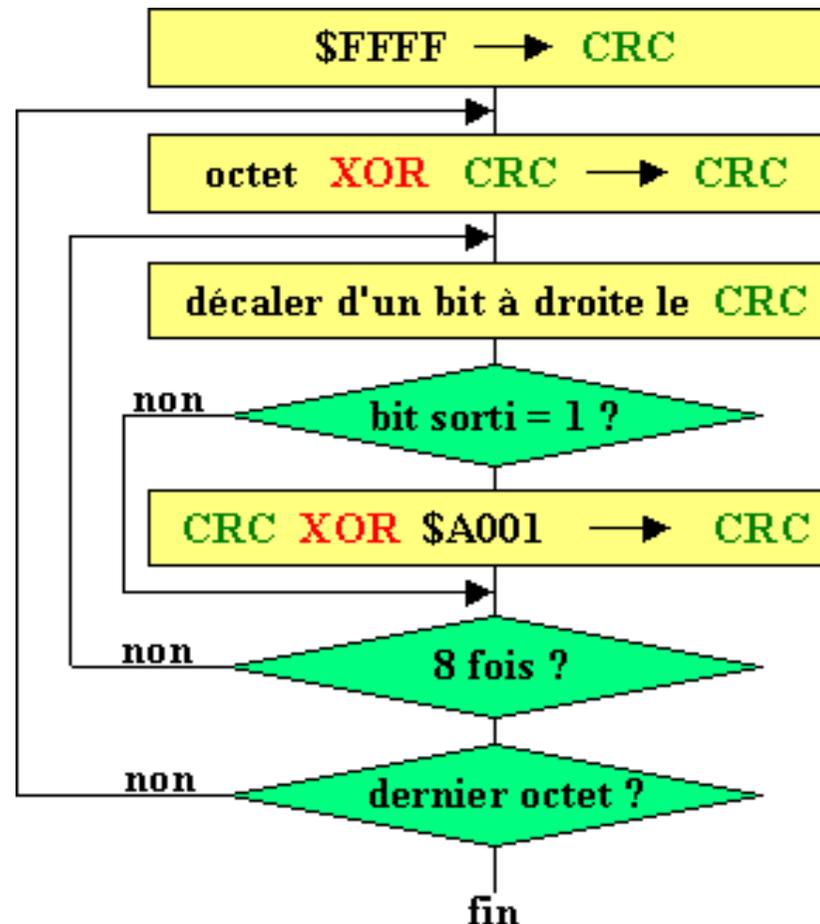
Code	Nature des fonctions MODBUS	TSX 37
H'01'	Lecture de n bits de sortie consécutifs	*
H'02'	Lecture de n bits de sortie consécutifs	*
H'03'	Lecture de n mots de sortie consécutifs	*
H'04'	Lecture de n mots consécutifs d'entrée	*
H'05'	Ecriture de 1 bit de sortie	*
H'06'	Ecriture de 1 mot de sortie	*
H'07'	Lecture du statut d'exception	
H'08'	Accès aux compteurs de diagnostic	
H'09'	Téléchargement, télé déchargement et mode de marche	
H'0A'	Demande de CR de fonctionnement	
H'0B'	Lecture du compteur d'événements	*
H'0C'	Lecture des événements de connexion	*
H'0D'	Téléchargement, télé déchargement et mode de marche	
H'0E'	Demande de CR de fonctionnement	
H'0F'	Ecriture de n bits de sortie	*
H'10'	Ecriture de n mots de sortie	*
H'11'	Lecture d'identification	*
H'12'	Téléchargement, télé déchargement et mode de marche	
H'13'	Reset de l'esclave après erreur non recouverte	



Modbus (5)

Le CRC (Cyclical Redundancy Check) est calculé par l'émetteur avant d'être transmis. Le récepteur calcule aussi le CRC et le compare avec le CRC reçu : des valeurs différentes indiqueront une erreur dans la transmission du message.

Le CRC, codé sur 2 octets (16 bits), est basé sur un OU EXCLUSIF (XOR) et se calcule de la façon suivante :



Profibus (1)

Profibus (Process Field Bus) est le nom d'un bus de terrain inventé par Siemens et devenu peu à peu une norme de communication dans le monde de l'industrie.

Le bus PROFIBUS-DP (Decentralised Peripheral) (périphérie décentralisée) est utilisé pour la commande de capteurs, d'actionneurs ou d'automates programmables par une commande centrale.

Il est utilisé aussi pour la connexion d'une « intelligence distribuée », c'est-à-dire la mise en communication de plusieurs automates les uns avec les autres. Les débits peuvent atteindre 12 MBit/s sur paires torsadées ou fibre optique.

La majorité des automates Siemens disposent d'une interface Profibus-DP pour le dialogue avec le PC de programmation.

Profibus respecte le modèle OSI (niveaux 1,2 et 7).



Profibus (2)

On reconnaît facilement un réseau Profibus-DP à la couleur de son câble : violet. En l'ouvrant, on distingue 2 fils : un vert et un rouge, nommé "A" et "B". Connecteur habituellement utilisé : DB9.

Comme de nombreux bus de terrain, le Profibus-DP s'appuie sur une liaison RS485. La vitesse généralement utilisée est de 12000kb/s, soit 12Mb/s (sur 100m).

Nombre maximum de stations par segment : 32

Profibus est un réseau multimaître :

- maître : pilote la transmission de données sous réserve d'avoir le droit d'accès au réseau (cf. jeton)*
- esclave : pas le droit d'accès au réseau. Acquiesce le message reçu ou transmet des données sur demande du maître*



Profibus (3)

Profibus utilise :

- la **méthode du jeton** pour les communications inter-maître : garantit l'accès au bus au moins une fois à chaque maître dans un temps donné. Le jeton est un droit de parole.
- la **méthode maître/esclave** : permet au maître possédant le jeton d'accéder à ses esclaves.



Jbus

Jbus est un réseau local industriel (bus de terrain), variante de Modbus, créé par April (gamme d'automate programmable industriel et réalisé par les sociétés Renault et Merlin Gerin).



Bus CAN (1)

CAN est un standard de fait développé par Bosh et Intel (1985) et qui respecte le modèle OSI (1,2). Le niveau application a été défini par ailleurs (cf. CANOpen).

CAN a été initialement été développé pour l'industrie automobile mais est aujourd'hui utilisé pour l'automatisme et les applications de contrôle. CAN est à ranger dans la catégorie des bus de terrain.

*Le protocole CAN est basé sur le principe de **diffusion générale** : lors de transmission, aucune station n'est adressée en particulier, mais le contenu de chaque message est explicité par une identification reçue par tous les abonnés. Grâce à cet identificateur, les noeuds, qui sont en permanence à l'écoute du réseau, reconnaissent et traitent les messages qui les concernent. Elles ignorent simplement les autres.*

CAN a connu un essor important : 11 millions de noeuds en 1996 et 149 millions de noeuds en 2001 !

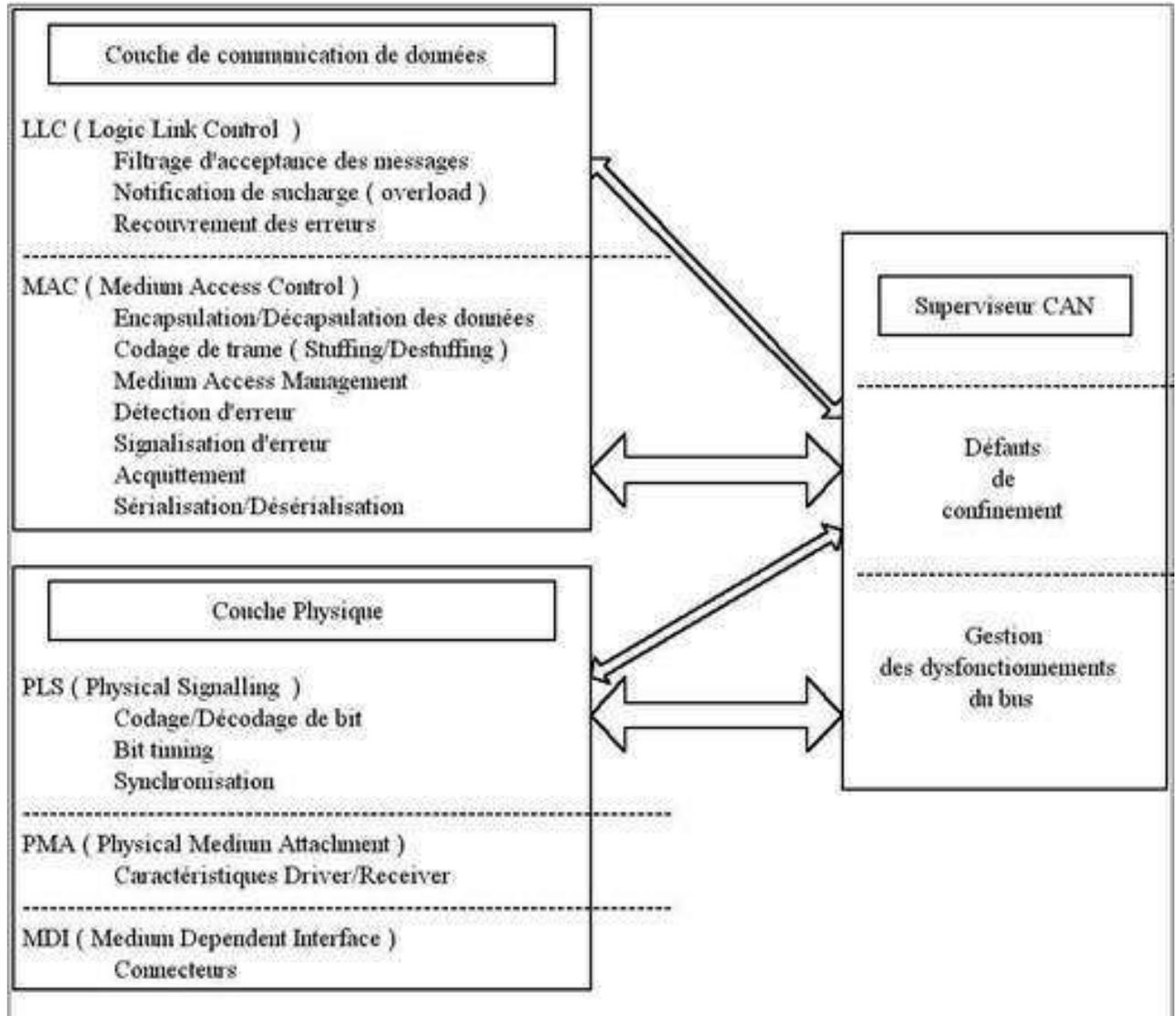


Bus CAN (2)

Modèle OSI :

Le bus CAN couvre la couche 2 (LIAISON) et la couche 1 (PHYSIQUE) du modèle OSI.

Remarque : la couche 7 (APPLICATION) peut être implémentée sous forme d'un protocole supplémentaire (CANOpen par exemple).



Bus CAN (3)

Couche physique :

Une paire torsadée blindée ou non

*Codage NRZ binaire (bit **dominant** : 0 et bit **récessif** : 1)*

Nombre max de noeuds : théoriquement suivant la taille du champ d'identification, pratiquement < 120 (suivant le circuit employé)

Débit brut de 20 kb/s à 1 Mb/s suivant la longueur du réseau

Topologie : bus

Couche liaison :

Communication multimaître et Arbitrage de type CSMA/CA

Arbitrage sur le champ d'identificateur de la trame (message)

L'entête (identificateur) de la trame donne sa priorité (identificateur faible = priorité forte)

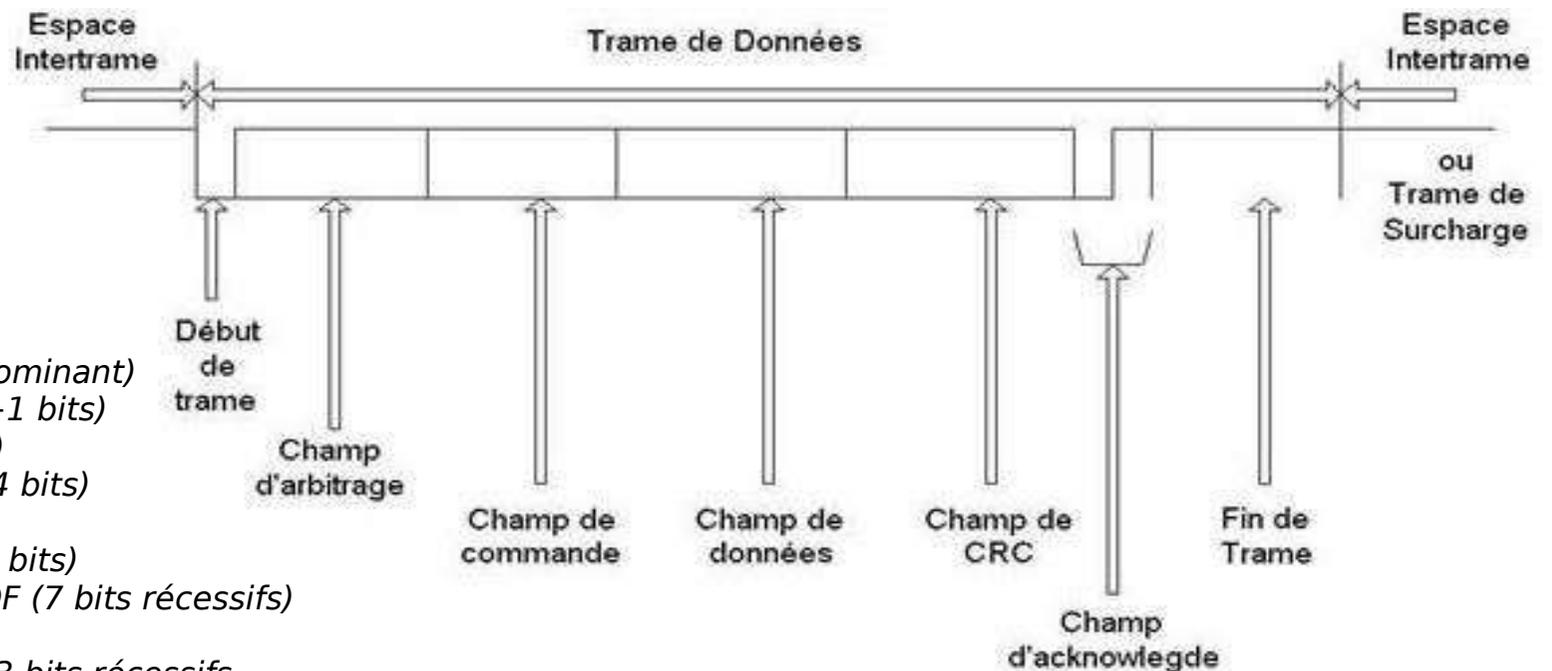
Deux trames définies : «Standard CAN» 2.0A avec un ID sur 11 bits (2032 noeuds en théorie) et «Extended CAN» 2.0B avec un ID sur 29 bits (536870912 noeuds en théorie)



Bus CAN (4)

Couche liaison (suite) : quatre types de trame :

- «data frame» (trame de données, prioritaire sur les «remote frame»)
- «remote frame» (trame de demande d'émission de données par un noeud),
- «error frame» (trame d'erreur) et
- «overload frame» (trame de surcharge)



Trame standard :

- début de trame SOF (1 bit dominant)
- + champ d'arbitrage (id 11+1 bits)
- + champ de contrôle (6 bits)
- + champ de données (0 à 64 bits)
- + CRC (16 bits)
- + champ d'acquiescement (2 bits)
- + champ de fin de trame EOF (7 bits récessifs)

espace inter-trame (IFS) de 3 bits récessifs



Bus CAN (5)

Méthode d'accès : le bus CAN utilise **CSMA/CR** (Carrier Sense Multiple Access / Collision Resolution) dont les principes généraux sont les suivants :

- Toutes les stations sont égales
- Chaque station émet quand elle veut (bus libre)
- Les collisions sont acceptées et sont détectées par les stations
- Les collisions sont gérées par le protocole
- Il y a une stratégie d'arbitrage des collisions
- La station qui remporte l'arbitrage continue sa transmission

Arbitrage : le problème de l'arbitrage résulte du fonctionnement multi maître. Si deux nœuds ou plus tentent d'émettre un message sur un bus libre, il faut régler les conflits d'accès. On effectue alors un arbitrage bit à bit non destructif (un ET) tout au long du contenu de l'identificateur. Ce mécanisme garantit qu'il n'y aura ni perte de temps, ni perte d'informations. Lorsqu'un bit récessif (1) est envoyé et qu'un bit dominant (0) est observé sur le bus, l'unité considérée perd l'arbitrage, doit "se taire" et ne plus envoyer aucun bit.

