



Le réseau CAN

Philippe Hoppenot

hoppenot@cemif.univ-evry.fr

<http://lsc.cemif.univ-evry.fr:8080/~hoppenot/presentationfrancaise.html>

Ce cours sur le réseau CAN est dispensé en seconde année d'IUT GEII. Son utilisation est libre avec les contraintes suivantes :

- Ne pas l'utiliser à des fins commerciales
- Citer la source lors de son utilisation
- Avertir l'auteur de son utilisation

Toutes les remarques sur le fond et la forme de ce document sont les bien venues.

Sommaire

I. Couche physique : 82C250	3
I.1. Caractéristiques	3
I.2. Données de référence de base	3
I.3. Diagramme fonctionnel	4
I.4. Brochage	4
I.5. Description fonctionnelle	4
I.6. Limites du circuit	5
II. Couche liaison : 82C200	5
II.1. Diagramme fonctionnel	5
<i>II.1.1. Interface Management Logic : IML</i>	<i>6</i>
<i>II.1.2. Transmit Buffer : TBF</i>	<i>6</i>
<i>II.1.3. Receive Buffer : RBF0 et RBF1</i>	<i>6</i>
<i>II.1.4. Bit Stream Processor : BSP</i>	<i>6</i>
<i>II.1.5. Bit Timing Logic : BTL</i>	<i>6</i>
<i>II.1.6. Transceiver Control Logic : TCL</i>	<i>6</i>
<i>II.1.7. Error Management Logic : EML</i>	<i>6</i>
<i>II.1.8. Controller Interface Logic : CIL</i>	<i>6</i>
II.2. Brochage	6
II.3. Registres de contrôle et tampons d'émission-réception	7
<i>II.3.1. Adressage</i>	<i>7</i>
<i>II.3.2. Zone de registres de contrôle</i>	<i>7</i>
<i>II.3.3. Tampon d'émission</i>	<i>8</i>
<i>II.3.4. Tampon de réception</i>	<i>9</i>
<i>II.3.5. Division de l'horloge de référence</i>	<i>9</i>
II.4. Protocole de communication	9
<i>II.4.1. Trame de données</i>	<i>9</i>
<i>II.4.2. Trame de demande de données</i>	<i>10</i>
<i>II.4.3. Trame d'erreur</i>	<i>10</i>
<i>II.4.4. Trame de surcharge</i>	<i>10</i>
<i>II.4.5. Espace inter-trame</i>	<i>10</i>
<i>II.4.6. Organisation du bus</i>	<i>11</i>
<i>II.4.7. Détection d'erreur</i>	<i>12</i>

Ce document présente le bus CAN dans son mode basic à partir du 82C200. La version actuelle est le SJA1000 qui fonctionne sur le même principe en ajoutant des fonctionnalités, en particulier la gestion intégrée d'une file d'attente en émission et en réception.

Au début des années 80, l'industrie automobile se trouve confrontée à l'augmentation du câblage dans les véhicules : nombreux capteurs et actionneurs (moteur, freins, embrayage...). La longueur des câbles utilisés peut atteindre plusieurs kilomètres avec une masse de plusieurs dizaines de kilogrammes. La nécessité d'un bus se fait sentir : 2 fils et l'alimentation 12V. Trois contraintes sont mises en avant : la sécurité, la vitesse (temps réel) et le coût.

1985 : Création de CAN par Intel et Bosch.

1989 : Premiers circuits disponible sur le marché.

Utilisé d'abord sur les voitures, il s'étend vite à d'autres secteurs. 1990, l'industrie textile. Puis les machines agricoles, les équipements navals, les appareils médicaux, les ascenseurs...

3 couches : Couche physique (82C250), couche liaison (82C200) et couche application (TD).

I. Couche physique : 82C250

82C250 : interface entre le contrôleur CAN et le bus physique.

I.1. Caractéristiques

Compatible *ISO/DIS11898*

Vitesse rapide (1 Mbaud)

Lignes protégées contre les perturbations de l'environnement

Contrôle de la pente des signaux pour limiter les interférences haute fréquence

Réception différentielle pour limiter les interférences électromagnétiques

Protections thermiques

Protection contre les courts-circuits

Mode d'attente faible consommation

Un composant non alimenté ne perturbe pas la ligne

Au moins 110 composants peuvent être connectés

I.2. Données de référence de base

Symbole	Paramètre	Min	Max	Unité
V_{CC}	Alimentation	4.5	5.5	V
I_{CC}	Alimentation		170	μA
$1/t_{bit}$	Vitesse max de transmission	1	-	MBaud
V_{CAN}	CANH, CANL niveau d'entrée/sortie	-8	+18	V
ΔV	Niveau différentiel	1.5	3.0	V
t_{pd}	Délais de propagation	-	50	ns
T_{amb}	Température de fonctionnement	-40	+125	$^{\circ}C$

Table 1 : Données de référence 82C250.

I.3. Diagramme fonctionnel

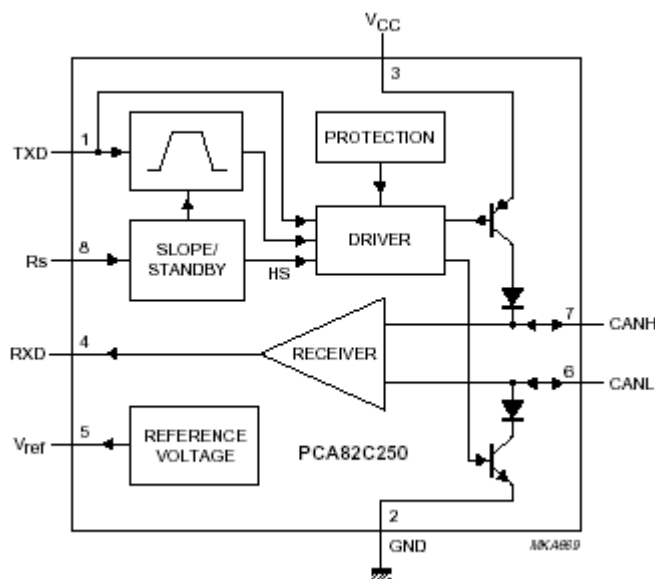


Figure 1 : Diagramme fonctionnel 82C250.

I.4. Brochage

SYMBOL	PIN	DESCRIPTION
TXD	1	transmit data input
GND	2	ground
V _{CC}	3	supply voltage
RXD	4	receive data output
V _{ref}	5	reference voltage output
CANL	6	LOW-level CAN voltage input/output
CANH	7	HIGH-level CAN voltage input/output
Rs	8	slope resistor input

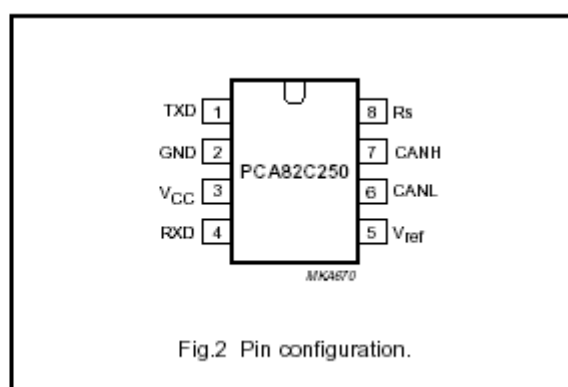


Fig.2 Pin configuration.

Figure 2 : Brochage 82C250.

I.5. Description fonctionnelle

Le PCA82C250 est l'interface entre le contrôleur de protocole CAN et le bus (couche physique). Sa première fonction est la gestion de la grande vitesse (jusqu'à 1mbaud) dans l'automobile. Le circuit propose une émission et une réception différentielles. Il est compatible avec la norme ISO/DIS11898.

Une limitation en courant protège l'étage de sortie de l'émetteur contre le court-circuit (avec l'alimentation positive ou négative). Bien que la dissipation d'énergie soit augmentée dans un fonctionnement avec erreur, cette limitation évitera la destruction de l'étage de sortie de l'émetteur.

Si une température de jonction excède 160°C, le courant limite des deux sorties d'émission est diminué. L'émetteur étant responsable de la majeure partie de la dissipation d'énergie, cela réduit cette dissipation et la température du circuit avec. Toutes les autres fonctions du contrôleur restent actives. La protection thermique est particulièrement utile lors d'un court-circuit sur la ligne du bus.

Les lignes CANH et CANL sont aussi protégées contre les perturbations électriques qui peuvent arriver dans les environnements automobiles.

La broche 8 (Rs) permet de sélectionner trois différents modes de fonctionnement : grande vitesse, contrôle du temps de transition, attente.

Dans le mode grande vitesse, les transistors de sortie sont simplement commandés fermés ou ouverts aussi vite que possible. Aucune mesure n'est prise pour limiter les temps de montée et de descente des signaux. L'utilisation de câbles blindés est recommandée pour éviter les perturbations de haute fréquence. Le mode rapide est sélectionné en connectant la broche 8 à la masse.

Pour des vitesses plus faibles ou des longueurs de bus plus petites, une paire torsadée ou parallèle non blindée peut être utilisée pour le bus. Pour réduire les perturbations haute fréquence, les temps de montée et de descente des signaux doivent être limités. Ils peuvent être contrôlés grâce à une résistance câblée entre la broche 8 et la masse. Le temps est alors proportionnel au courant de sortie de cette broche.

Si un niveau haut est appliqué à la broche 8, le circuit entre en mode d'attente faible consommation. L'émetteur est éteint et le récepteur est en veille (faible courant). Si des bits dominants sont détectés (tension différentielle sur le bus > 0.9V), RxD est positionnée au niveau bas. Le processeur doit réagir à cet état en remettant l'émetteur en position normale (via la broche 8). Le récepteur étant en mode faible courant lors d'un état de veille, le premier message reçu est perdu.

I.6.Limites du circuit

Symbole	Paramètre	Min.	Max.	Unité
V_{CC}	Alimentation	-0.3	+9.0	V
V_n	Tension sur les broches 1, 4, 5 et 8	-0.3	$V_{CC}+0.3$	V
$V_{6,7}$	Tension sur les broches 6 et 7	-8.0	+18.0	V
T_{stg}	Température de stockage	-55	+150	°C
T_{amb}	Température de fonctionnement	-40	+125	°C

Table 2 : Limites du circuit 82C250.

II.Couche liaison : 82C200

II.1.Diagramme fonctionnel

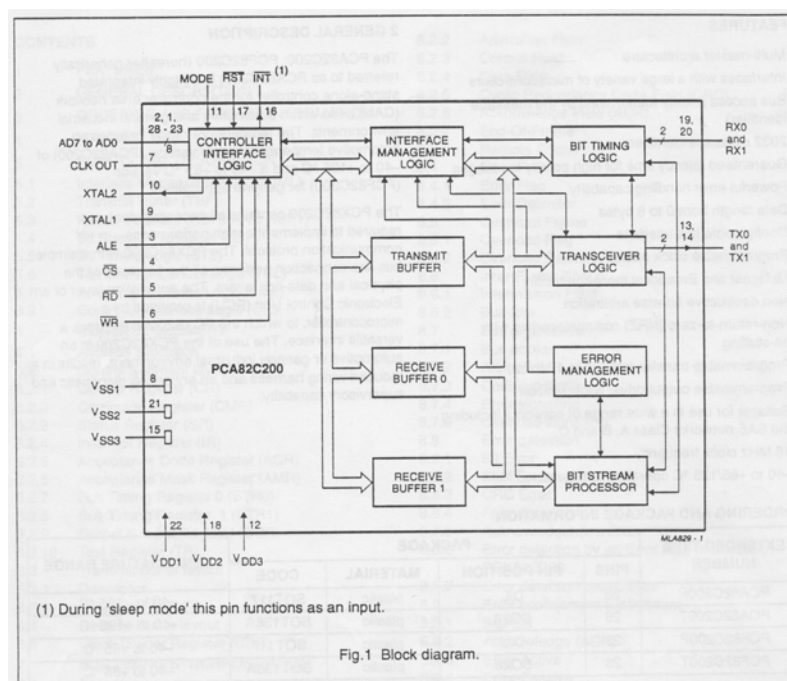


Figure 3 : Diagramme fonctionnel 82C200.

II.1.1. Interface Management Logic : IML

L'IML interprète les commandes du microprocesseur, alloue les tampons (TBF, RBF0 et RBF1) et produit les interruptions et les informations de statuts à l'attention du μ P.

II.1.2. Transmit Buffer : TBF

Le TBF est une zone mémoire de 10 octets dans laquelle le μ P écrit les messages à émettre sur le réseau.

II.1.3. Receive Buffer : RBF0 et RBF1

Les RBF0 et RBF1 sont deux zones mémoire de 10 octets chacune qui sont utilisées à tour de rôle pour stocker les messages provenant du réseau. Un message peut être lu par le μ P pendant qu'un second est reçu.

II.1.4. Bit Stream Processor : BSP

Le BSP est un séquenceur contrôlant le flux des données entre le tampon d'émission, les tampons de réception et le bus.

II.1.5. Bit Timing Logic : BTL

Le BTL synchronise le circuit sur le flot de données du bus.

II.1.6. Transceiver Control Logic : TCL

Le TCL contrôle le driver de sortie.

II.1.7. Error Management Logic : EML

L'EML gère les erreurs de protocole du bus.

II.1.8. Controller Interface Logic : CIL

Le CIL est l'interface avec le μ P externe.

II.2. Brochage

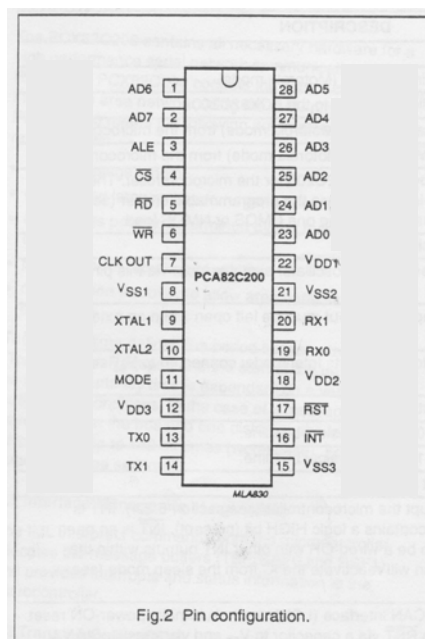


Figure 4 : Brochage 82C200.

II.3. Registres de contrôle et tampons d'émission-réception

II.3.1. Adressage

La zone mémoire interne du 82C200 se compose d'une zone de registres de contrôle et d'une zone de tampons d'émission-réception. Elle contient 31 registres. Les registres de contrôle sont utilisés à l'initialisation du circuit pour configurer les paramètres de communication. Ils permettent aussi de contrôler l'émission et la réception de messages. La zone de tampon permet de spécifier les messages à émettre ou de lire eux reçus.

TITLE	ADDR	7	6	5	4	3	2	1	0
Control Segment									
Control Register	0	Test Mode	Sync	reserved	Overrun Interrupt Enable	Error Interrupt Enable	Transmit Interrupt Enable	Receive Interrupt Enable	Reset Request
Command Register	1	reserved	reserved	reserved	Goto Sleep	Clear Overrun Status	Release Receive Buffer	Abort Transmission	Transmission Request
Status Register	2	Bus Status	Error Status	Transmit Status	Receive Status	Transmission Complete Status	Transmit Buffer Access	Data Overrun	Receive Buffer Stat
Interrupt Register	3	reserved	reserved	reserved	Wake-Up Interrupt	Overrun Interrupt	Error Interrupt	Transmit Interrupt	Receive Interrupt
Acceptance Code Register	4	AC.7	AC.6	AC.5	AC.4	AC.3	AC.2	AC.1	AC.0
Acceptance Mask Register	5	AM.7	AM.6	AM.5	AM.4	AM.3	AM.2	AM.1	AM.0
Bus Timing Register 0	6	SJW.1	SJW.0	BRP.5	BRP.4	BRP.3	BRP.2	BRP.1	BRP.0
Bus Timing Register 1	7	SAM	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0
Output Control Register	8	OCTP1	OCTN1	OCPOL1	OCTP0	OCTN0	OCPOL0	OCMODE1	OCMODE0
Test Register (note 1)	9	reserved	reserved	Map Internal Register	Connect RX Buffer 0 CPU	Connect TX Buffer CPU	Access Internal Bus	Normal RAM Connect	Float Output Driver
Transmit Buffer									
Identifier	10	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5	ID.4	ID.3
RTR, Data Length Code	11	ID.2	ID.1	ID.0	RTR	DLC.3	DLC.2	DLC.1	DLC.0
bytes 1-8	12-19	Data	Data	Data	Data	Data	Data	Data	Data

Table 3 : Registres du 82C200.

II.3.2. Zone de registres de contrôle

II.3.2.1. Control Register : CR

Utilisé pour changer le comportement du 82C200, il est accessible en lecture/écriture.

II.3.2.2. Command Register : CMR

Utilisé pour initier des actions du 82C200, il est accessible en écriture seule..

II.3.2.3. Status Register : SR

Utilisé pour connaître l'état du 82C200, il est accessible en lecture seule.

II.3.2.4. Interrupt Register : IR

Utilisé pour identifier une source d'interruption, il est en lecture seule. Il est remis à 0 à chaque lecture.

II.3.2.5. Acceptance Code Register : ACR

Il s'agit d'un filtre de réception des messages. Il est accessible en lecture/écriture quand le bit Reset Request est à 1.

Equation partielle de test : (ID.10...ID.3) = (AC.7...AC.0).

II.3.2.6. Acceptance Mask Register : AMR

Masque d'évaluation de ACR.

Equation de test total : [(ID.10...ID.3) = (AC.7...AC.0)] ou (AM7...AM.0) = 1111 1111.

Exercice : 3 postes. 1 reçoit tout, les 2 autres reçoivent certains messages.

II.3.2.7. Bus Timing Register 0 : BTR0

Permet de définir une base pour la vitesse de transmission et un temps de synchronisation. Ces temps doivent être égaux sur tous les systèmes d'un même réseau.

Base de la vitesse : $t_{SCL} = 2t_{CLK}(32BPR.5 + 16BPR.4 + 8BPR.3 + 4BPR.2 + 2BPR.1 + BPR.0 + 1)$.
 t_{CLK} = horloge du PCX82C200.

Temps de synchronisation : $t_{SJW} = t_{SCL}(2SLW.1 + SJW.0 + 1)$.

II.3.2.8. Bus Timing Register 1 : BTR1

Permet de définir la durée de chaque bit, le point d'échantillonnage et le nombre d'échantillonnages.

Durée d'un bit : $t_{TSEG1} = t_{SCL}(8TSEG1.3 + 4TSEG1.2 + 2TSEG1.1 + TSEG1.0 + 1)$.

Point d'échantillonnage : $t_{TSEG2} = t_{SCL}(4TSEGS2.2 + 2TSEGS2.1 + TSEGS2.0 + 1)$.

Nombre d'échantillonnages : SAM=1 3 échantillonnages (vitesses moyenne et faible).

SAM=0 1 échantillonnage (vitesse rapide).

II.3.2.9. Output Control Register : OCR

Choix de différentes configurations du driver de sortie.

4 modes (OCMODE1 et OCMODE2) : Normal, avec horloge en sortie, bi-phase, test.

Types de sorties (OCTPx, OCTNx, OCPOLx) : Float, pull-down, pull-up, push-pull.

II.3.2.10. Test Register : TR

Utilisé pour les tests de production seulement.

II.3.3. Tampon d'émission

II.3.3.1. Descripteur : DSCRx

Identificateur (ID) : 8+3=11 bits d'identification de message dont les 8 de poids fort sont mis en correspondance avec le "acceptance code".

Demande de transmission (RTR) : HIGH : Trame de demande.

LOW : Trame de données.

Longueur de message (DLC) : Nombre d'octets du message. Doit être inférieur ou égal à 8.

II.3.3.2. Champs de données

8 octets de données au maximum.

II.3.4. Tampon de réception

Il a la même structure que le tampon d'émission.

II.3.5. Division de l'horloge de référence

Permet de fournir au processeur pilote une horloge de fréquence à choisir.

II.4. Protocole de communication

Différentes trames : trames de données ; trames de demande de données ; trames d'erreur ; trames de surcharge.

Les trames de données et de demande de données sont générées par l'application. Les trames d'erreur et de surcharge générées par le circuit lui-même avec signal à l'application (Control Register).

Des temps entre ces différentes trames doivent être respectés.

II.4.1. Trame de données

Elle se compose des champs de bits de la Figure 5.

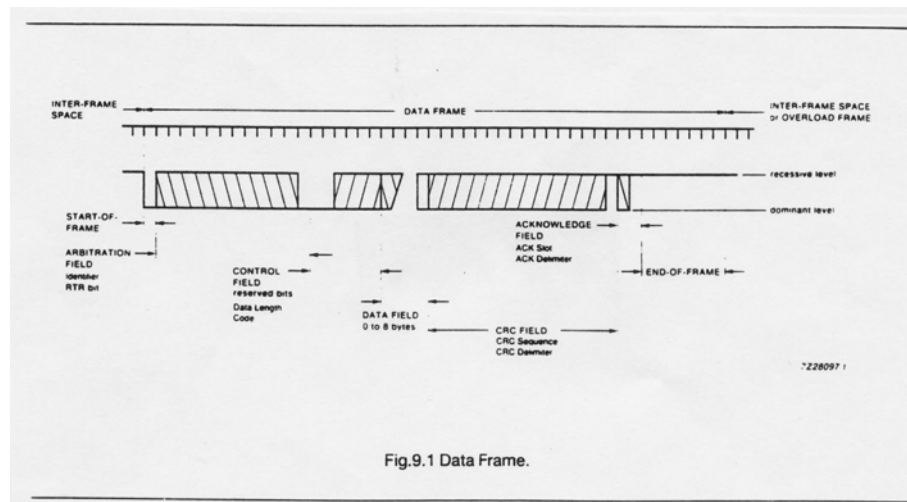


Figure 5 : Trame de données.

II.4.1.1.- Début de trame

1 bit de synchronisation matérielle.

II.4.1.2.- Champ d'arbitrage

ID + RTR : Arbitrage en cas d'accès multiple au bus. Bit récessif (haut), bit dominant (bas).

ID : Identification du message (ID.10 à ID.0).

Attention : ID.10 à ID.4 ne peuvent pas être récessifs tous en même temps (conflit avec les messages d'erreur).

RTR : Demande de données non prioritaire (RTR haut : récessif). Pas d'erreur générée.

II.4.1.3.- Champ de contrôle

2 bits réservés + 4 bits pour la longueur des données (0 à 8 inclus).

II.4.1.4.- Champs de données

de 0 à 8 octets.

II.4.1.5.- CRC

15 bits de code de redondance et 1 bit de fin de CRC. Etudier pour une trame de 127 bits au maximum.

II.4.1.6.- Acquiescement

2 bits : Acknowledge Slot et Acknowledge Delimiter. l'émetteur positionne l'Acknowledge Slot à un niveau récessif. Chaque récepteur le met à un niveau dominant. Ainsi, l'émetteur sait qu'au moins un récepteur a reçu le message correctement (sans erreur : CRC).

II.4.1.7.- Fin de trame

7 bits récessifs (2 de + que pendant l'émission de la trame elle-même).

II.4.2. Trame de demande de données

Ressemble à la trame de données à trois exceptions près :

- RTR récessif
- Longueur des données ignorée
- Pas de données

II.4.3. Trame d'erreur

2 champs : drapeau d'erreur et délimiteur d'erreur.

II.4.3.1. Drapeau d'erreur

6 bits de même polarité consécutifs => violation de la loi des bits de bourrage (5 bits de même polarité consécutifs au maximum).

Erreur active ou passive suivant que les bits sont récessifs ou dominants.

II.4.3.2. Délimiteur d'erreur

8 bits récessifs.

II.4.4. Trame de surcharge

2 champs : drapeau de surcharge (6 bits dominants) et Délimiteur de surcharge (8 bits récessifs).

2 conditions de déclenchement :

- le récepteur n'est pas prêt,
- réception de bits dominants pendant un temps de repos du bus (Intermission Field).

2 départ possibles :

- 1^{er} bit d'un champ d'attente (Intermission Field),
- 1 bit après la détection d'un bit dominant pendant un champ d'attente.

Deux trames de surcharge au maximum peuvent être envoyées à la suite.

II.4.5. Espace inter-trame

Temps d'attente entre différentes trames.

II.4.5.1.Champ intermission

3 bits récessifs entre les trames de données et de demande de données.

II.4.5.2.Attente du bus

Attente d'une nouvelle transmission (bus à l'état récessif). Un niveau dominant sur le bus rompt l'attente.

II.4.6.Organisation du bus

II.4.6.1.Accès au bus

Une transmission commence pendant une attente du bus. Elle est repérée par un bit de début de trame (synchronisation matérielle, dominant).

II.4.6.2.Arbitrage du bus

Si plusieurs systèmes accèdent au réseau en même temps, l'arbitrage est réalisé grâce à l'identificateur (11 bits) du message et au bit RTR. Chaque système émetteur écoute le bus en même temps qu'il émet. S'il lit un bit dominant alors qu'il avait positionné un bit récessif, il stoppe sa transmission. L'identificateur est envoyé le premier, poids fort en tête, puis de RTR.

Exemple : ID1 = 001 1101 1111 RTR = 0 => Dominant

ID2 = 001 1110 0000 RTR = 1

Attention : un ID ne peut pas commencer par 7 bits récessifs (compatibilité avec d'autre contrôleur CAN).

II.4.6.3.Codage/décodage

Les cinq champs suivants sont codés en utilisant des bits de bourrage :

- Début de trame
- Arbitrage
- Contrôle
- Données
- CRC

Quand 5 bits de même polarités sont détectés, le système rajoute un bit de bourrage de l'autre polarité. Les trames d'erreur et de surcharge ont un statut particulier : elles ne suivent pas ce codage.

II.4.6.4.Signalisation d'erreur

Cinq types d'erreur peuvent être détectés. En cas d'erreur de bit, de bourrage, de format ou d'acquiescement la transmission d'un drapeau d'erreur commence au bit suivant. En cas d'erreur de CRC, la transmission d'un drapeau d'erreur commence après le champ d'acquiescement du message.

Un drapeau d'erreur viole la loi du bourrage et celle du format des trames. Chaque système qui détecte une telle erreur transmet à son tour un drapeau d'erreur.

Après la transmission d'un drapeau d'erreur, chaque système écoute le bus jusqu'à détecter une transmission de niveau dominant-récessif. A cet instant, chaque système a fini de transmettre son drapeau d'erreur et transmet 7 bits récessifs (délimiteur d'erreur).

L'échange interrompu peut alors reprendre avec la rémission automatique de la trame.

II.4.6.5.Signalisation de surcharge

De même format que la trame d'erreur, celle de surcharge est traitée différemment. Elle n'initie pas la rémission de la trame.

II.4.7.Détection d'erreur

Cinq types d'erreur sont détectés.

II.4.7.1.Erreur de bit

Chaque transmetteur écoute le bus en même temps qu'il y écrit. S'il lit un bit de niveau différent de celui qu'il a émis, une erreur est signalée. Deux exception :

- Pendant l'arbitrage, un bit récessif peut être écrasé par un bit dominant.
- Pendant l'acquittement, seuls les récepteurs peuvent reconnaître une erreur.

II.4.7.2.Erreur de bourrage

Deux cas possibles. Plus de cinq bits consécutifs de même polarité : erreur de bourrage. Une perturbation transforme un des cinq bits de même polarité qui ont généré un bit de bourrage. Le bit de bourrage n'est donc pas reconnu comme tel et l'erreur de bourrage ne peut pas être déclenchée. D'autres détections d'erreur sont alors possibles (bit, CRC, format).

II.4.7.3.Erreur de CRC

CRC + parité. Attributs :

- 127 bits maximum.
- 112 bits significatifs maximum (83 avec le 82C200).
- CRC avec un polynôme de degré 15
$$f(X) = (X^{14} + X^9 + X^8 + X^6 + X^5 + X^4 + X^2 + X + 1)(X + 1)$$
soit 1100 0101 1001 1001.
- Distance de Hamming : 6.

II.4.7.4.Erreur de format

Elle intervient pour la 4 champs suivants :

- Fin de trame.
- Intermission.
- Délimiteur d'acquittement.
- Délimiteur de CRC.

Une erreur correspond à la détection d'un bit dominant au lieu d'un bit récessif.

II.4.7.5.Erreur d'acquittement

Détectée par l'émetteur quand il ne reçoit pas de bit dominant à la place du récessif mis dans le bit Acknowledge Slot.

II.4.7.6.Signalisation de l'erreur

Deux compteurs signalent la présence d'erreur de bus, l'un en émission, l'autre en réception. Ils s'incrémentent de 8 en cas d'erreur et se décrémentent de 1 en cas de succès. Quand une valeur de 96 est atteinte, l'erreur de statut et l'interruption d'erreur sont positionnées. Quand ils atteignent 255, le bus est mis en bus-off. Il faut alors procéder à un reset du système pour remettre les compteurs à 0 ainsi que les bits signalant les erreurs.

II.4.7.7.Capacité de détection

Le CRC garantit un taux d'erreur inférieur à 3.10^{-5} .