

CAN

Controller Area Network

Nel 1986 la Bosch, su richiesta del gruppo Mercedes e BMW, cercò di risolvere gran parte dei problemi che si riscontravano nella diffusione delle informazioni tra i sempre più numerosi dispositivi elettronici presenti all'interno di un'autovettura (air conditioning control, antiblock braking system, traction control etc.).

Fu così che nacque il protocollo di rete **CAN BOSCH** 

Il **CAN** (acronimo di **CONTROLLER AREA NETWORK**) è un protocollo di comunicazione seriale che assicura un elevato livello di sicurezza per controlli realtime.

Il suo dominio applicativo spazia dalle reti ad alta velocità fino ad impianti elettronici di basso costo.

Nel campo dell'automazione industriale motori, sensori e numerosi altri dispositivi elettronici sono connessi tramite CAN (con bit rate fino a 1 Mbit/s).

Allo scopo di ottenere un modello trasparente ed un'implementazione flessibile, si potrebbe inquadrare il protocollo CAN in 3 differenti livelli:

- il livello di **OGGETTO**
- il livello di **TRASFERIMENTO**
- il livello **FISICO**

LIVELLO DI OGGETTO

- filtraggio dei messaggi
- interpretazione dei messaggi

LIVELLO DI TRASFERIMENTO

- isolamento dei guasti
- rilevamento di errori
- segnalazione di errori
- acknowledgement
- arbitraggio
- suddivisione in pacchetti
- sincronizzazione

LIVELLO FISICO

- rappresentazione binaria
- trasmissione dell'informazione

Ai primi due livelli (livello di **OGGETTO e di **TRASFERIMENTO**) sono associati tutti i servizi e le funzioni del livello **DATA-LINK** definito nel modello **ISO/OSI**.**

Lo scopo principale del livello **DATA-LINK è di trasformare una trasmissione grezza in una linea per il livello superiore che appaia libera da errori di trasmissione non segnalati.**

Visto che il livello **FISICO accetta e trasmette sequenze di bit senza far riferimento al loro significato o alla loro struttura, è compito del livello data-link creare e riconoscere i limiti dei pacchetti.**

Il livello di **OGGETTO è in grado inoltre di:**

- capire quali messaggi sono stati trasmessi**
- decidere quali messaggi ricevuti dal livello di trasferimento vanno processati**

L'obiettivo del livello di **TRASFERIMENTO** è principalmente il controllo dei pacchetti, il rilevamento di errori e l'invio di messaggi di errore.

All'interno del livello di trasferimento si valuta sia quando il bus è libero per cominciare una nuova trasmissione sia quando dare inizio alla fase di ricezione.

Il livello **FISICO** si occupa del trasferimento dei bit di informazione tra nodi differenti.

Evidentemente il livello fisico, all'interno di una rete, deve essere lo stesso per tutti i nodi.

ALCUNE PROPRIETÀ DEL CAN

Ecco alcune delle più interessanti **proprietà** del CAN:

- ✓ gestione di vari livelli di priorità associati ai singoli messaggi
- ✓ garanzia di tempi di latenza
- ✓ flessibilità di configurazione
- ✓ ricezione multicast con sincronizzazione temporale
- ✓ rilevamento e segnalazione di errori
- ✓ ritrasmissione automatica di messaggi errati
- ✓ gestione di errori temporanei
- ✓ gestione di stati di malfunzionamento permanente
- ✓ isolamento automatico di nodi difettosi

CONFIGURAZIONE RETE

- Un nodo CAN non usa informazioni sulla **configurazione** del sistema
- possono essere **aggiunti nodi** senza che vengano richiesti cambiamenti a livello hardware o software (**flessibilità del sistema**);

MESSAGGI

- il contenuto dei messaggi è caratterizzato da un **identificatore** che non indica l'indirizzo destinazione del pacchetto ma descrive semplicemente quanto vi è contenuto, affinché tutti i nodi della rete siano in grado di decidere attraverso il **filtraggio del messaggio** se "appropriarsene" o meno;
- la **priorità** è stabilita dal "tipo" di identificatore

ACCESSO AL CANALE

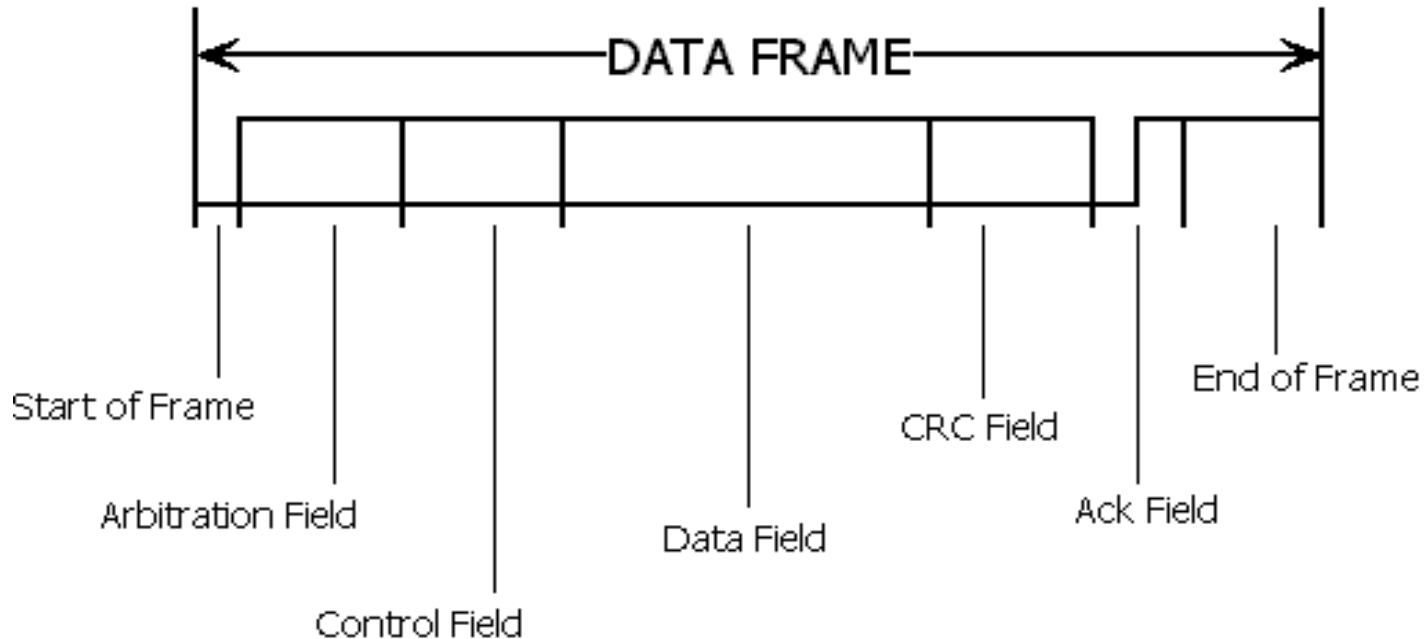
- Quando il bus è libero ogni unità è abilitata a trasmettere informazioni (**multimaster**).
- Solo l'unità a cui è associato il messaggio con priorità più alta si guadagna l'**accesso al canale**.
- Durante la trasmissione contemporanea di un **bit 1** ed un **bit 0**, il valore riscontrabile sul canale sarà **0**. Di conseguenza d'ora in poi si parlerà di bit '**dominante**' (bit 0) e bit '**recessivo**' (bit 1).

REMOTE FRAME

- Attraverso l'invio di un **REMOTE FRAME**, un nodo in attesa di informazioni può richiedere il **DATA FRAME** contenente i dati di interesse.

- Naturalmente il **DATA FRAME** ed il corrispondente **REMOTE FRAME** hanno lo stesso identificatore.

DATA FRAME: 7 segmenti

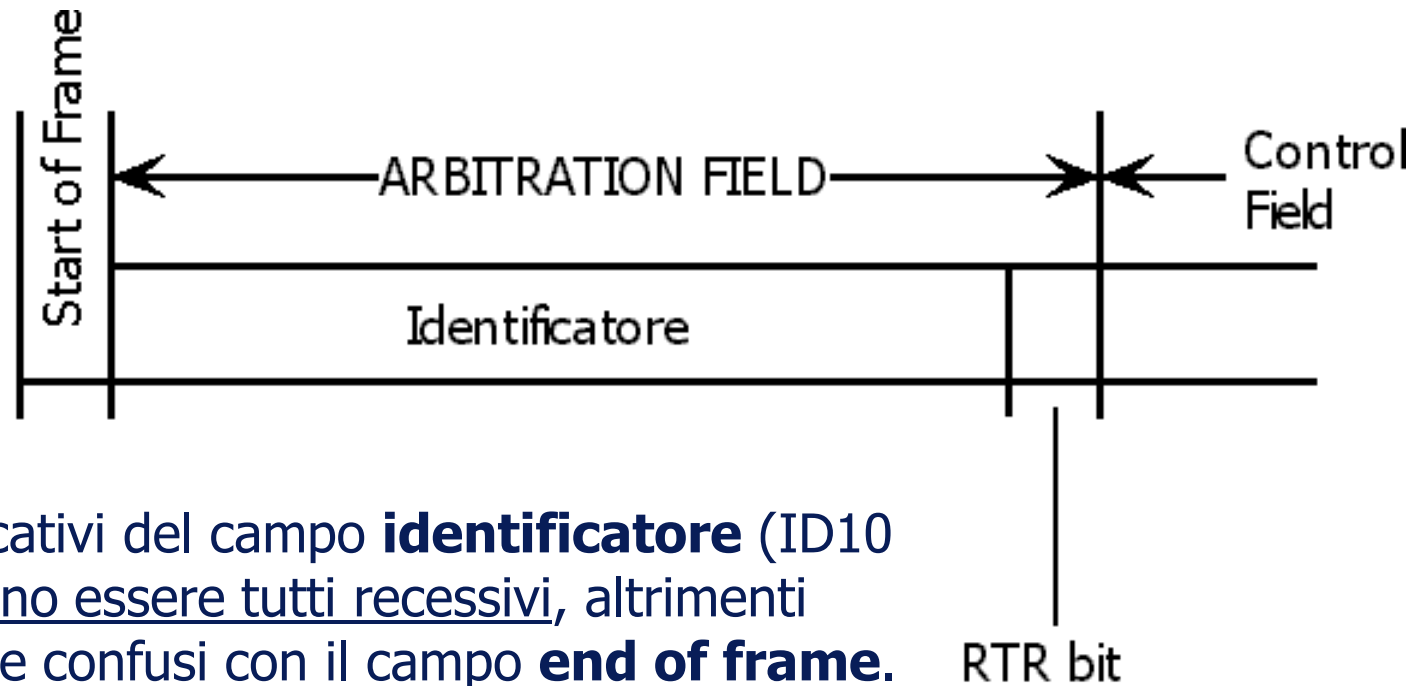


Il campo **START OF FRAME** marca con un singolo bit dominante l'inizio di ogni data frame e remote frame.

Il campo **END OF FRAME** è costituito da una sequenza di 7 bit recessivi, sia nei data frame che nei remote frame.

Arbitration field

comprende un campo **identificatore (11 bit)** ed un RTR bit (**Remote Transmission Request**).



I 7 bit più significativi del campo **identificatore** (ID10 - ID4) non possono essere tutti recessivi, altrimenti potrebbero essere confusi con il campo **end of frame**.

L'**RTR bit** assume valore dominante se si tratta di un data frame, valore recessivo, invece, nel caso di remote frame.

Arbitraggio

Quando 2 o più unità stanno trasmettendo contemporaneamente, il conflitto è risolto con un meccanismo di **ARBITRAGGIO che assicura che non siano perse né informazioni né tempo.**

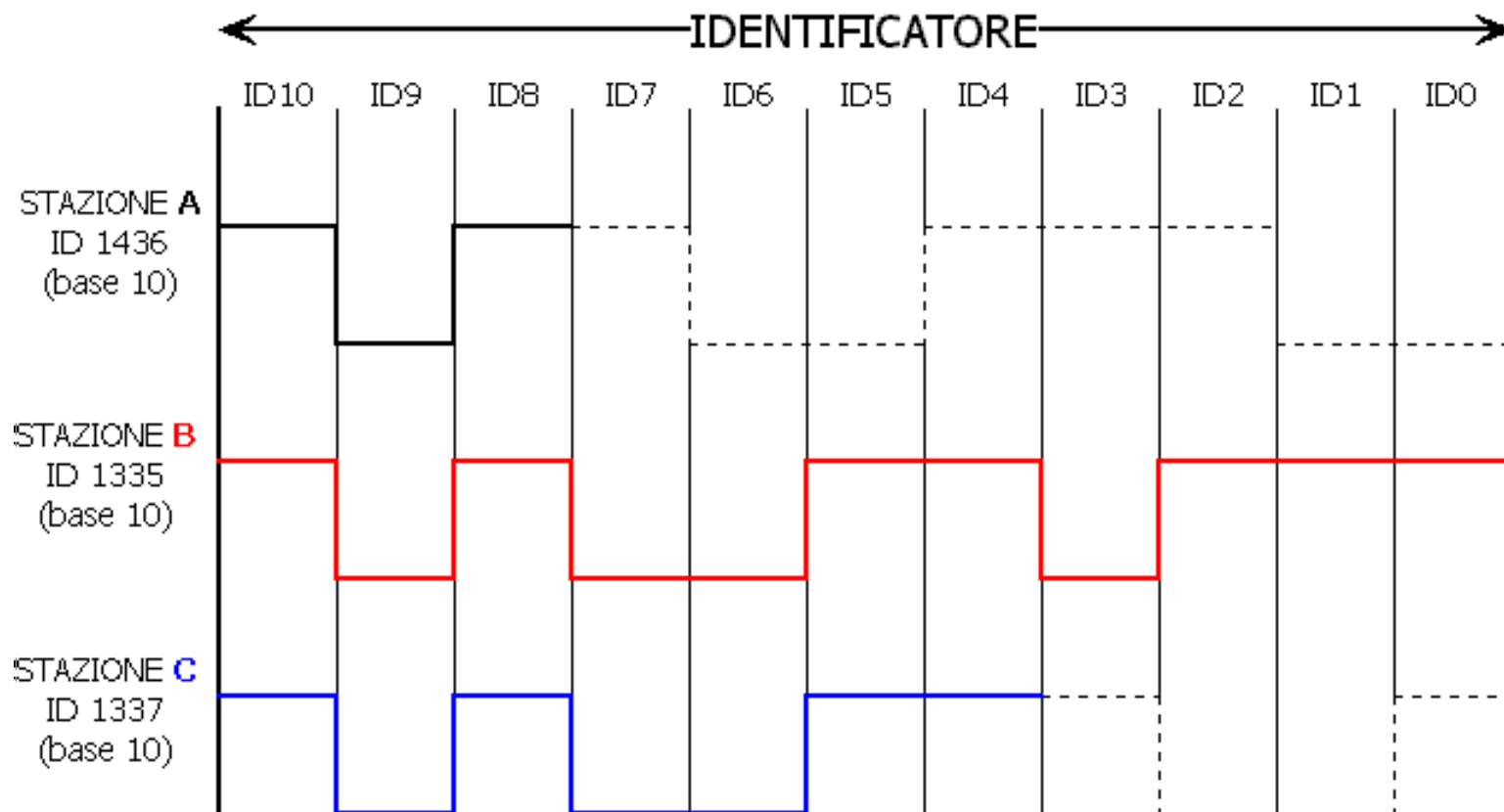
Se un data frame ed un remote frame con lo stesso identificatore sono trasmessi contemporaneamente, il data frame prevale sul remote frame.

Durante la fase di arbitraggio ogni trasmettitore confronta il livello del bit trasmesso con il livello monitorato sul canale.

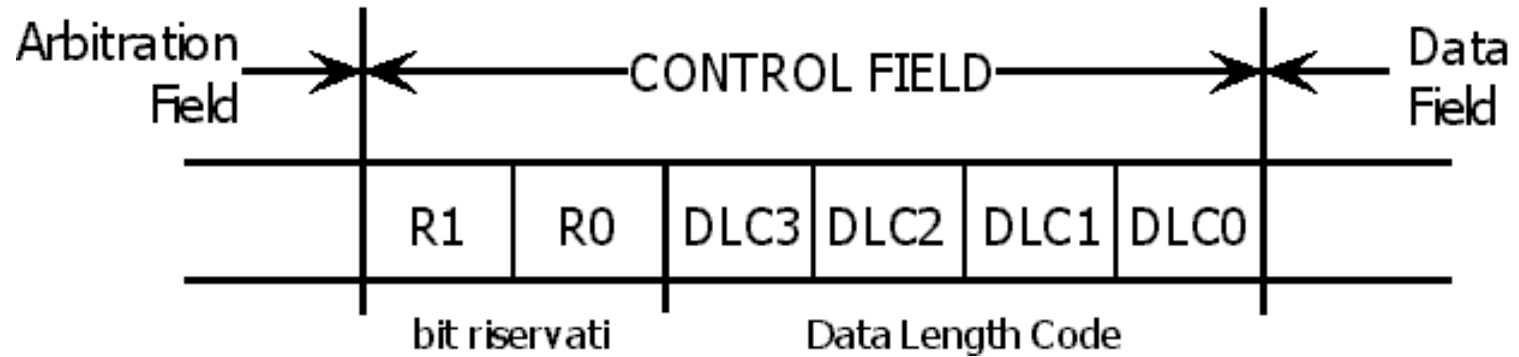
Se i 2 livelli coincidono l'unità continua a trasmettere.

Quando il livello associato al bit è recessivo mentre sul canale si riscontra un livello dominante, l'unità interrompe immediatamente la trasmissione.

Segue un esempio di **confitto** tra 3 stazioni che si contendono l'accesso al canale.



Control Field

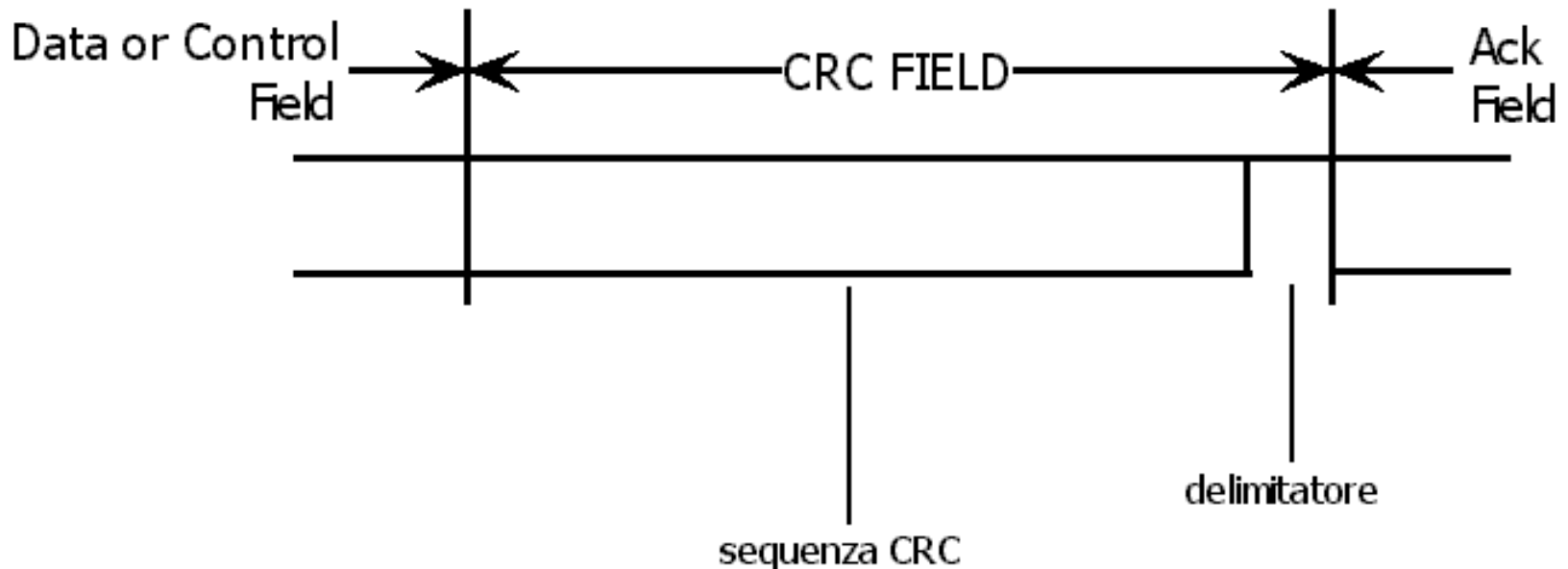


Il CONTROL FIELD occupa 6 bit, di cui 4 sono destinati al data length code e 2, entrambi dominanti, sono riservati per eventuali espansioni future.

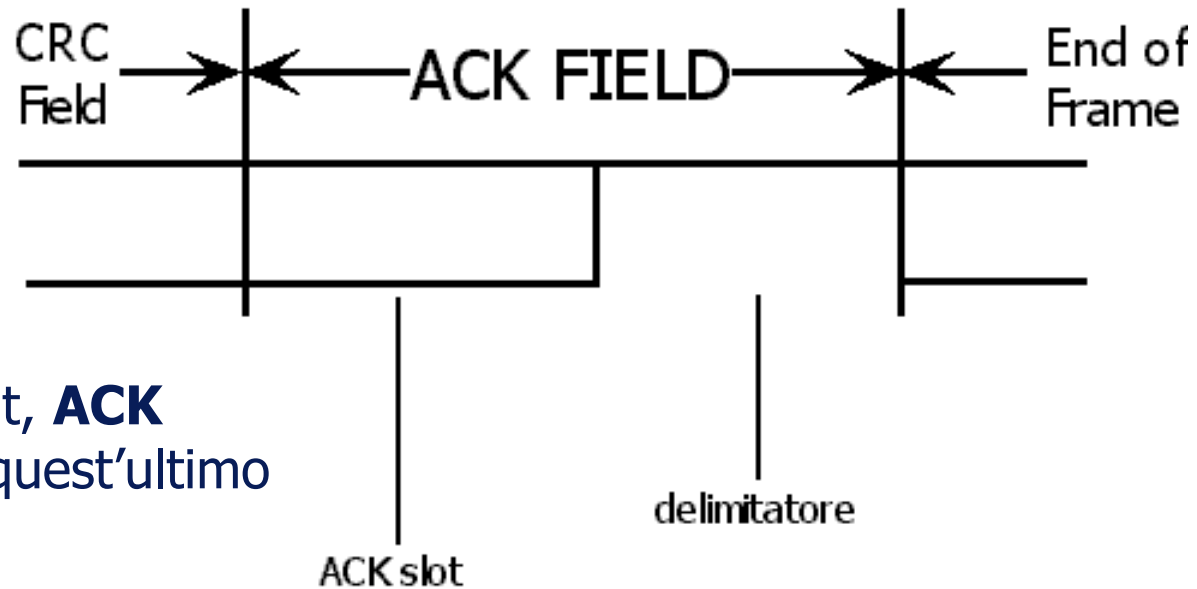
DATA Field e CRC Field

Il **DATA FIELD** contiene i dati da trasferire sul canale e può occupare da 0 ad 8 byte.

Il **CRC FIELD** comprende 16 bit, di cui i primi 15 costituiscono una sequenza di controllo ottenuta da un **codice a ridondanza ciclica** e l'ultimo bit (recessivo) fa da **delimitatore**.



ACK Field



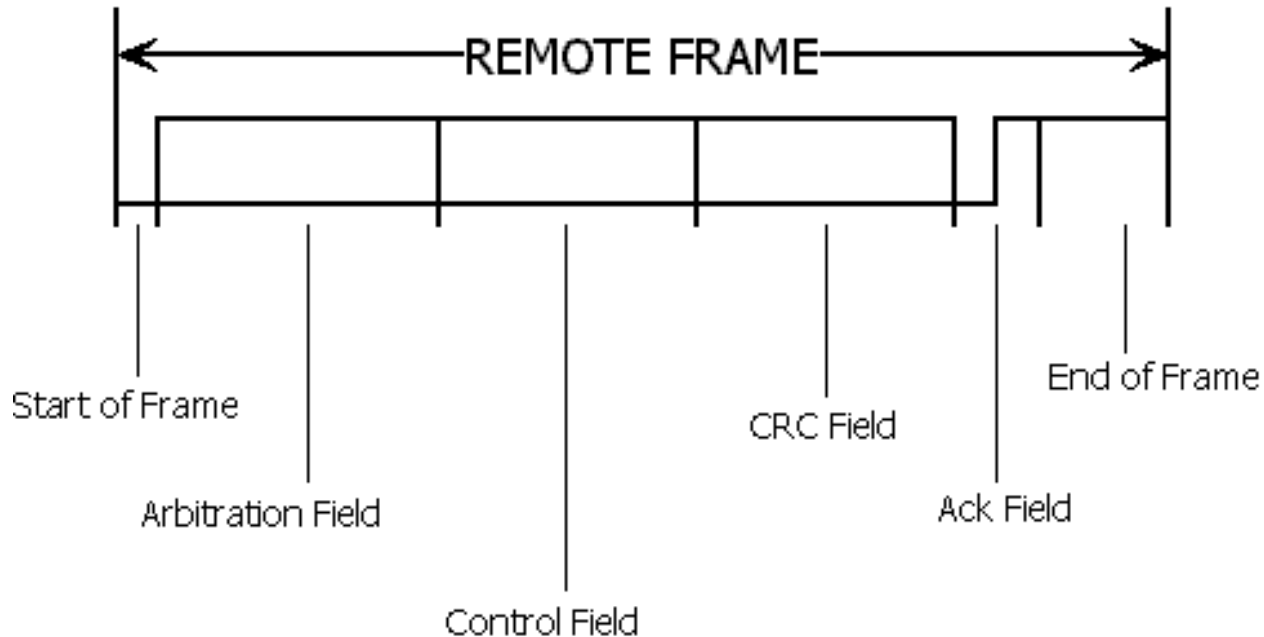
L'**ACK FIELD** occupa 2 bit, **ACK slot** ed **ACK delimiter**, quest'ultimo sempre recessivo.

Ogni stazione trasmittente invia un ACK slot contenente un bit recessivo.

Un ricevitore che abbia ricevuto correttamente un messaggio, comunica l'evento al trasmettitore ponendo a 0 il bit associato all'**ACK slot**.

Remote Frame

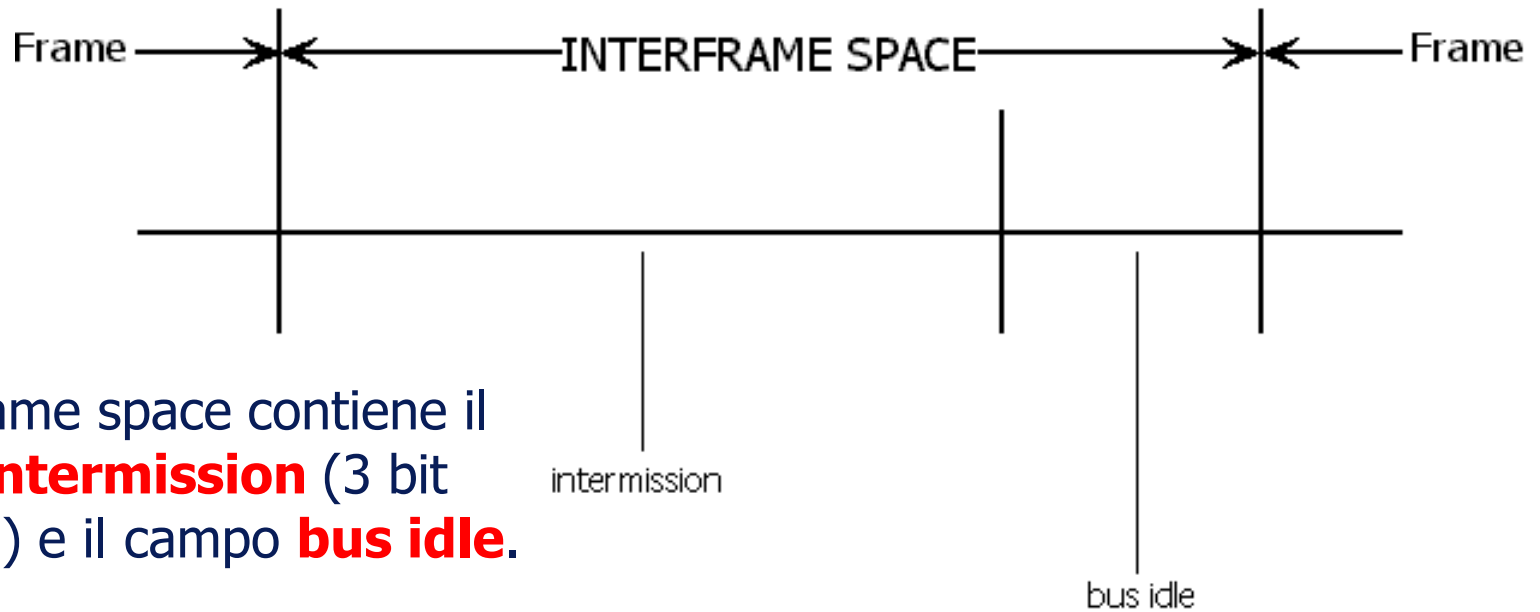
Un remote frame è a tutti gli effetti un data frame privo del data field.



Interframe Space

Data frame e remote frame **sono separati dai pacchetti che li precedono (data frame, remote frame, error frame, overload frame) da un campo di bit detto INTERFRAME SPACE.**

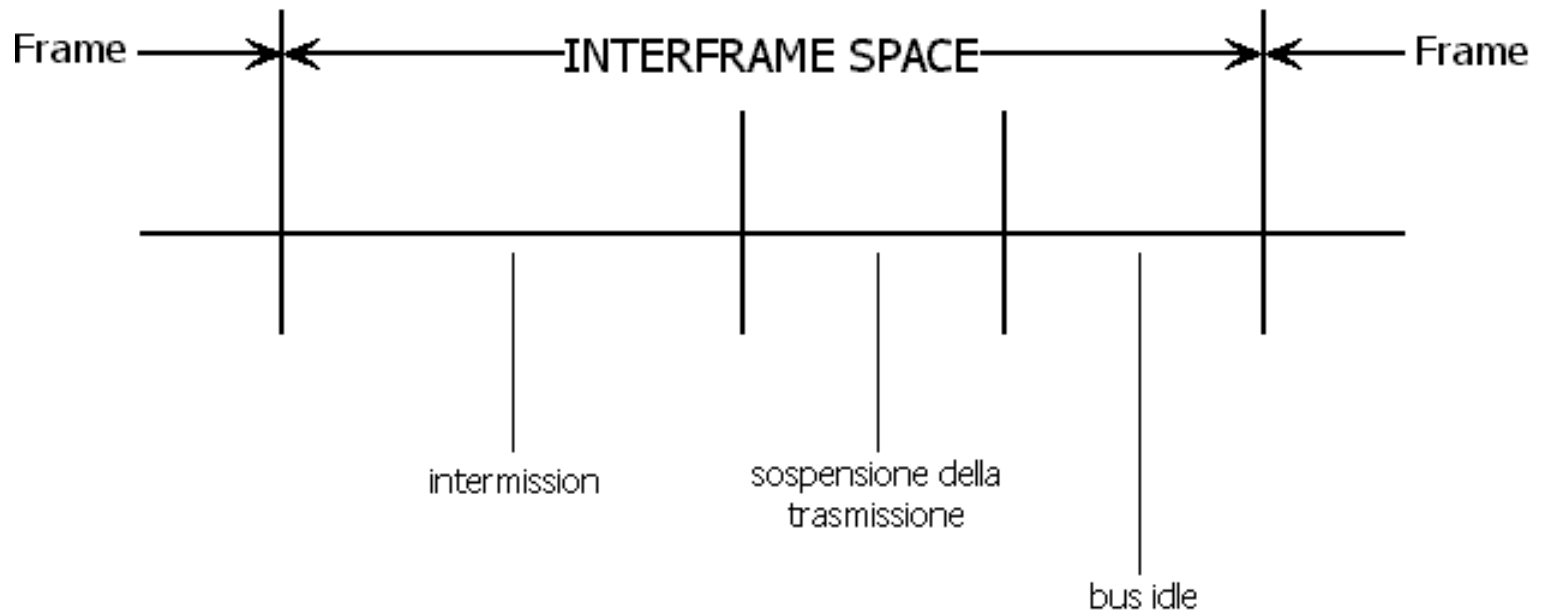
Error frame e overload frame **(anche multipli), invece, non sono preceduti da interframe space.**



L'interframe space contiene il campo **intermission** (3 bit recessivi) e il campo **bus idle**.

Suspended Transmission Field

Per le stazioni 'error passive' che hanno trasmesso il messaggio precedente esiste un ulteriore campo, detto **suspend transmission field**.



Durante la fase di **intermission** nessuna stazione è abilitata a cominciare la trasmissione di un eventuale data frame o remote frame.

L'unica azione possibile è la segnalazione di una **condizione di overload**.

Bus Idle

La fase di **bus idle** può essere di lunghezza arbitraria.

Ogni stazione che abbia qualcosa da trasmettere può accedere al bus.

Un messaggio, la cui trasmissione è stata interrotta durante l'immissione sulla rete di un ulteriore messaggio, comincia ad essere ritrasmesso a partire dal primo bit dopo l'intermission field.

Il rilevamento di un **bit dominante** sul canale è interpretato come **start of frame**.

Dopo aver trasmesso un messaggio, una **stazione 'error passive'** manda a seguito dell'intermission field (esattamente prima di iniziare a trasmettere un ulteriore messaggio o prima dell'eventuale spazio di bus idle) 8 bit recessivi.

La stazione è stata messa così in condizione di perdere la fase di arbitraggio e, se nel frattempo ha inizio una nuova trasmissione (causata da un'altra unità), la stazione diventerà automaticamente ricevente di tale messaggio.

Isolamento dei guasti

Per quanto concerne l'isolamento dei guasti, ogni stazione può trovarsi in 3 stati differenti:

1. **ERROR ACTIVE**
2. **ERROR PASSIVE**
3. **BUS OFF**

Una stazione **'error active'** al rilevamento dell'errore attiva l'**active error flag**.

Una stazione **'error passive'** comunica la presenza di un eventuale errore attivando il **passive error flag**.

Una stazione in stato di **'bus off'** non può più inviare informazioni sul canale.

Esistono 2 contatori a bordo di ogni stazione, entrambi responsabili di eventuali passaggi di stato:

- il **TEC (transmit error counter)**
- il **REC (receive error counter)**

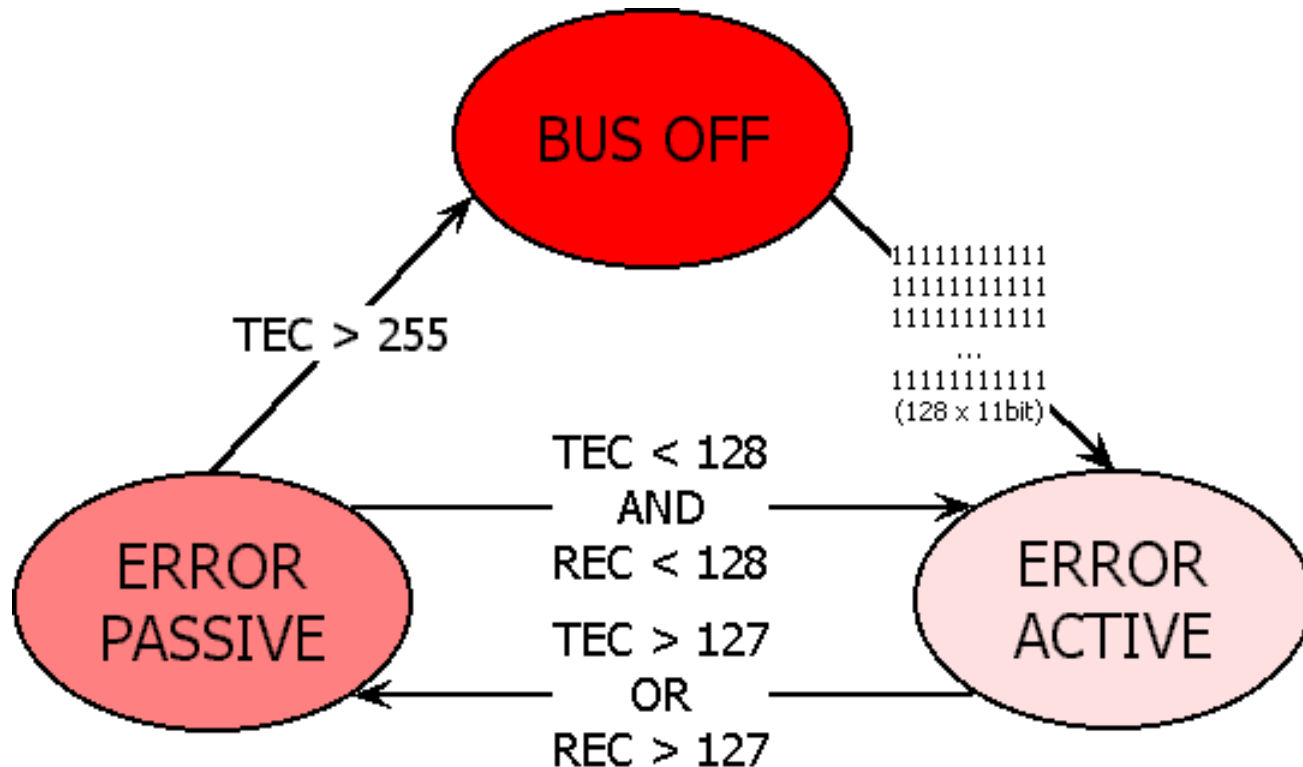
Regole di passaggio di stato

1. Quando un ricevitore rileva un errore il **REC** è incrementato di 1 unità, a meno che non si stia tramettendo un active error flag o un overload flag.
2. Quando un ricevitore rileva un bit dominante come primo bit dopo aver trasmesso un error flag, il **REC** sarà incrementato di 8 unità.
3. Quando un trasmettitore invia un error flag, il **TEC** viene incrementato di 8 unità.
4. Se un trasmettitore rileva un bit error mentre sta inviando un active error flag o un overload flag, il **TEC** è incrementato di 8 unità.
5. Se un ricevitore rileva un bit error mentre sta inviando un active error flag o un overload flag, il **REC** è incrementato di 8 unità.

- 6.** Ogni nodo tollera fino a 7 bit dominanti consecutivi dopo aver inviato un active/passive error flag o un overload flag.
Dopo il quattordicesimo bit dominante consecutivo (in caso di active flag o di overload flag) o dopo aver rilevato l'ottavo bit dominante consecutivo dopo un passive error flag o dopo ogni altra sequenza di 8 bit dominanti consecutivi, ogni trasmettitore ed ogni ricevitore incrementa rispettivamente il TEC ed il REC di 8 unità.
- 7.** Dopo una trasmissione riuscita di un messaggio (ack ricevuto e nessun errore rilevato) il **TEC** viene decrementato di 1 unità.
- 8.** Dopo aver ricevuto correttamente un messaggio, ogni stazione decrementa il proprio **REC** di 1 unità se il valore precedente era compreso tra 1 e 127, lo riporta, invece, ad un valore compreso tra 119 e 127 se in precedenza era maggiore di 127.
- 9.** Un nodo è **'error passive'** quando almeno uno tra **TEC e REC** è maggiore o uguale di 128.

10. Un nodo è in stato di 'bus off' quando il TEC è maggiore o uguale di 255.

11. Un nodo in stato di '**bus off**' può ritornare '**error active**' (con entrambi i contatori settati a zero) dopo che sono state rilevate sul canale 128 occorrenze di 11 bit recessivi consecutivi.



Active e Passive Error Flag

Una stazione 'error active' segnala una condizione di errore trasmettendo un active error flag subito dopo il CRC field, distruggendo così ACK field e end of frame.

Tutte le altre stazioni che si accorgono della condizione di errore segnalata danno inizio alla trasmissione del proprio error flag.

Per effetto della sovrapposizione degli error flag, la sequenza di bit dominanti rilevabile sul bus può variare da un minimo di 6 ad un massimo di 12 bit.

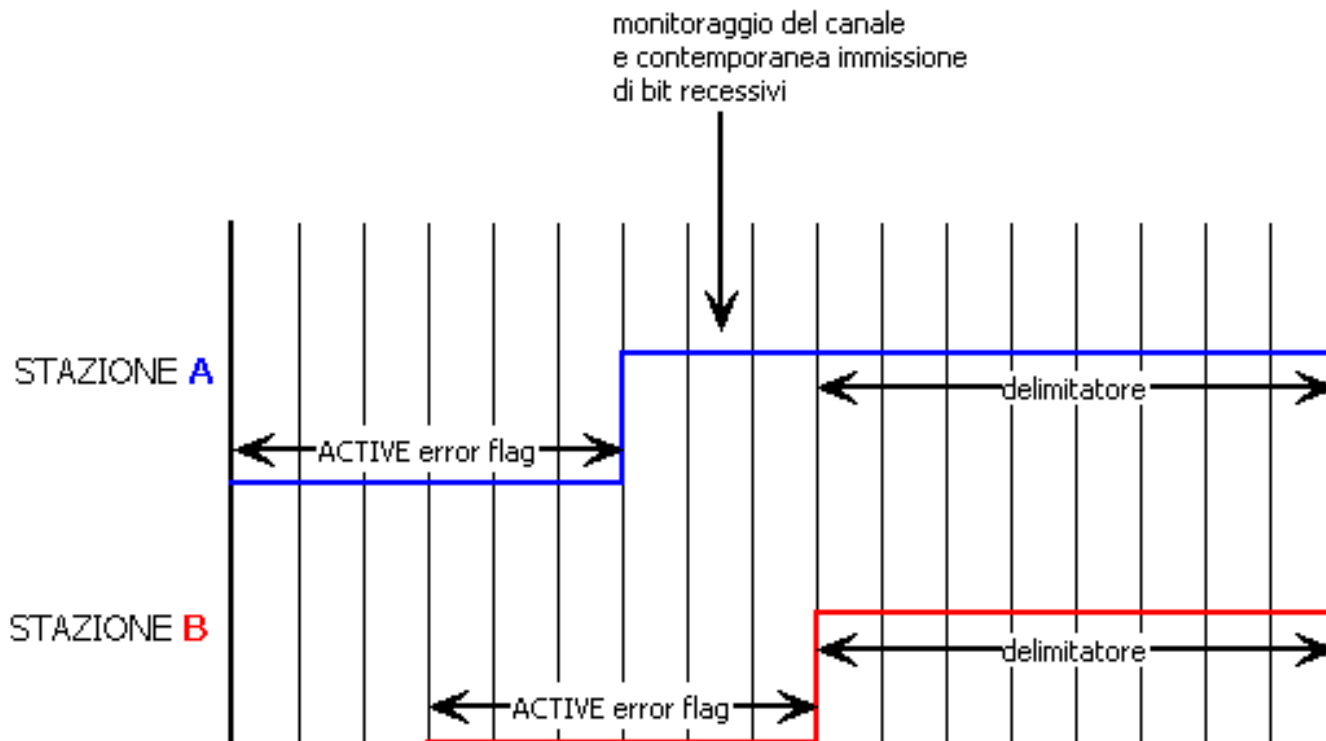
Una **stazione 'error passive'** tenta di segnalare una condizione di errore inviando un **passive error flag**.

L'unità 'error passive', a partire dall'inizio della sequenza di **passive error flag**, si aspetta di monitorare sul canale 6 bit consecutivi dello stesso valore.

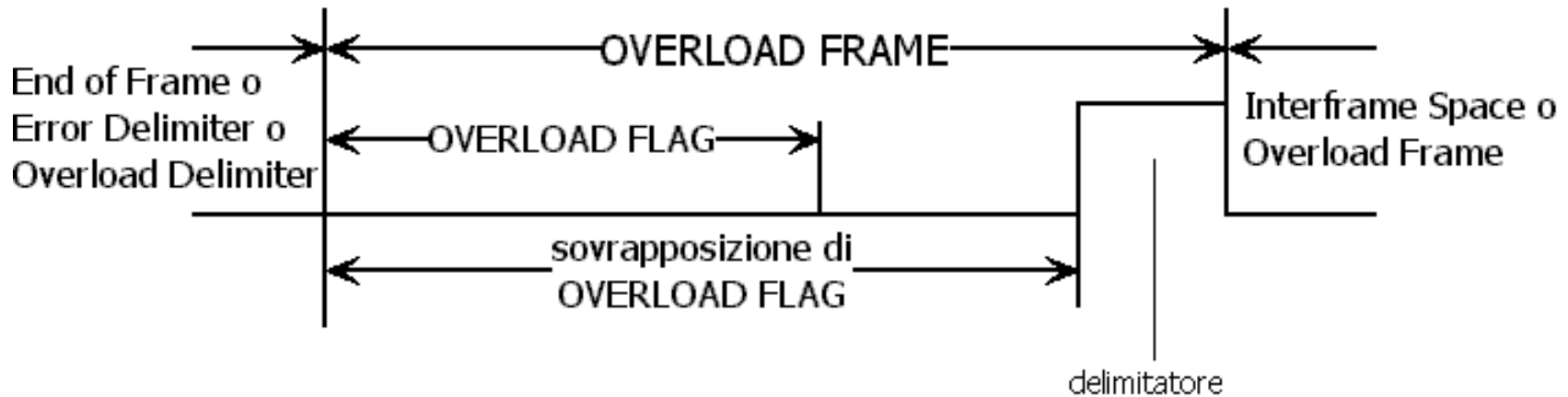
Questo evento assicura alla stazione che la condizione di errore sia stata opportunamente diffusa sul canale.

Il campo delimitatore è composto da 8 bit recessivi.

Dopo la trasmissione di un error flag, ogni stazione invia bit recessivi e attende il primo bit recessivo sul canale. A questo punto si completa la trasmissione inviando i rimanenti 7 bit recessivi.



Overload Frame (2 campi)



Ci sono 2 categorie **condizioni di overload**, che possono portare alla trasmissione di un overload flag:

1. le **condizioni interne di un ricevitore**, che necessita di un ritardo della trasmissione del successivo data/remote frame
2. Il rilevamento di un **bit dominante** durante la fase di **intermission** (caratteristica dell'**interframe space**)

Rilevamento Errori

Il meccanismo di rilevamento degli errori garantisce che siano rilevati:

- **tutti gli errori globali**
- **tutti gli errori locali del trasmettitore**
- **al massimo 5 errori distribuiti casualmente in un messaggio**
- **un qualsiasi numero dispari di errori in un messaggio**

La probabilità di non rilevare messaggi contenenti informazioni errate è solitamente minore del valore della seguente espressione:

$$4.7 \cdot 10^{-11} \cdot \text{MessageErrorRate}$$

Il tempo trascorso tra il rilevamento di un errore e la ritrasmissione del messaggio è di circa **29 bit time** (**bit time = 1 / bit rate**), se non ci sono altri errori.

Tipologie di Errori

I campi start of frame, arbitration field, control field, data field e CRC field sono codificati col metodo del BIT STUFFING.

Ogni volta che il trasmettitore rileva 5 bit consecutivi identici, automaticamente inserisce un bit complementare nella sequenza da trasmettere.

Il protocollo CAN prevede **5 tipologie di errore** riscontrabili (e non mutuamente esclusive):

1. **BIT ERROR**
2. **STUFF ERROR**
3. **CRC ERROR**
4. **FORM ERROR**
5. **ACKNOWLEDGEMENT ERROR**

Tipologie di Errori

BIT ERROR

Una stazione trasmette un bit ma legge sul bus un valore differente. Naturalmente questo tipo di errore non è riscontrato nella trasmissione di arbitration field, ack slot, passive error flag.

STUFF ERROR

È rilevato al sesto bit identico consecutivo in un campo che dovrebbe essere codificato con bit stuffing.

CRC ERROR

È rilevato dal ricevitore quando la sequenza CRC inviata dal trasmettitore risulta errata.

FORM ERROR

È rilevato quando un campo contiene valori illegali, quindi che non rispettano la forma fissata (ad esempio ack delimiter dominante anziché recessivo).

ACKNOWLEDGEMENT ERROR

È rilevato dal trasmettitore quando non riscontra sul canale un bit dominante all'interno del segmento ack slot (nessuna unità ha ricevuto il pacchetto).

Error Flag

Quando una stazione rileva un bit error, stuff error, form error o acknowledgement error, questa comincia immediatamente la trasmissione dell'error flag.

Se viene rilevato un CRC error, la trasmissione dell'error flag ha inizio subito dopo l'ack delimiter (a causa dei tempi di calcolo del CRC), a meno che un ulteriore error flag non sia già stato immesso sulla rete.

CAN 1.2

Per poter disporre di un campo di indirizzamento più ampio, passando così da **11 bit (CAN versione 1.1) a **29 bit**, recentemente è stata introdotta un'estensione del protocollo CAN (CAN versione 1.2).**

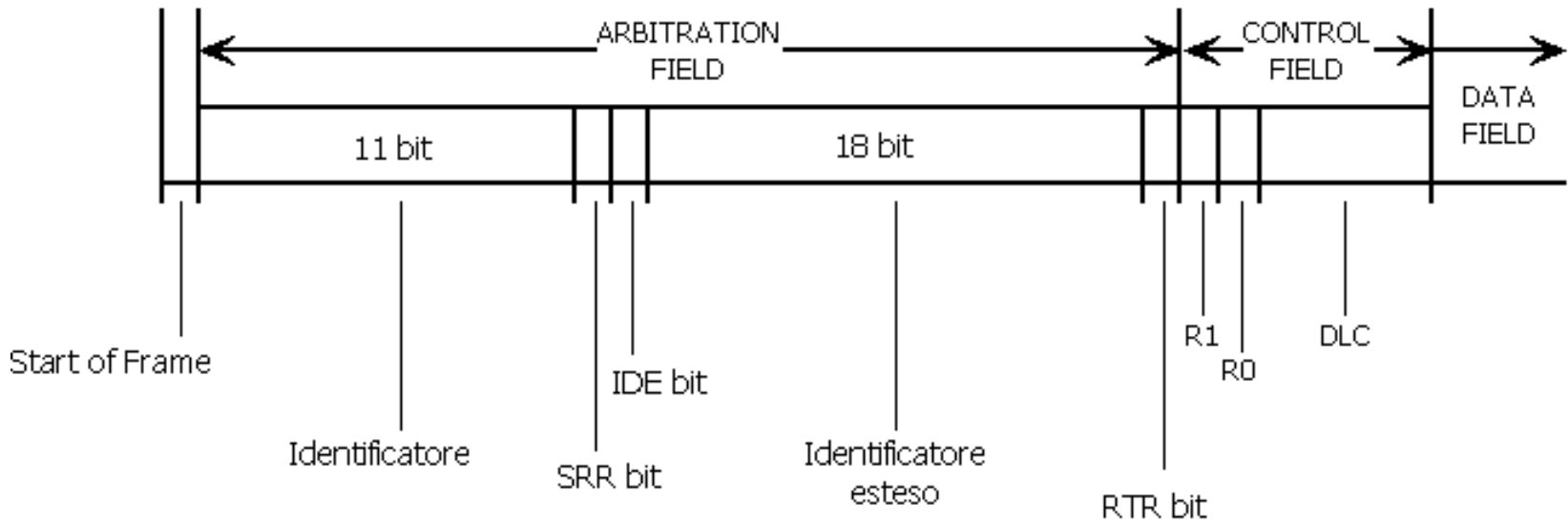
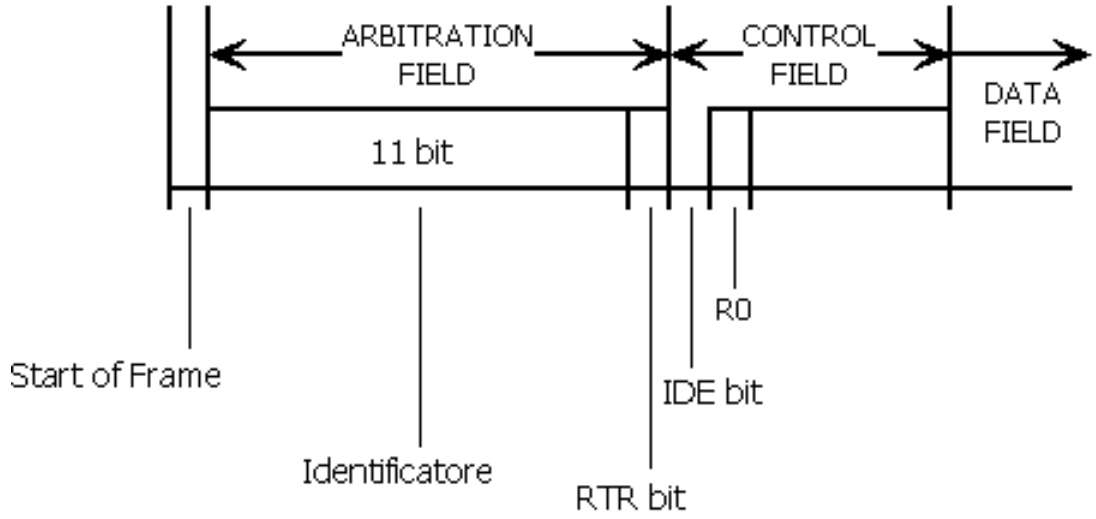
Il nuovo sistema di comunicazione è perfettamente compatibile con il precedente.

I livelli di **OGGETTO** e **TRASFERIMENTO** sono stati rinominati rispettivamente livello **LLC (Logical Link Control)** e livello **MAC (Medium Access Control)**, allineandosi così alla nomenclatura prevista dal modello ISO/OSI.

I DATA/REMOTE FRAME presentano 2 diversi formati:

- **STANDARD FRAME** (identificatore di 11 bit)
- **EXTENDED FRAME** (identificatore di 29 bit)

STANDARD/EXTENDED FRAME



EXTENDED FRAME

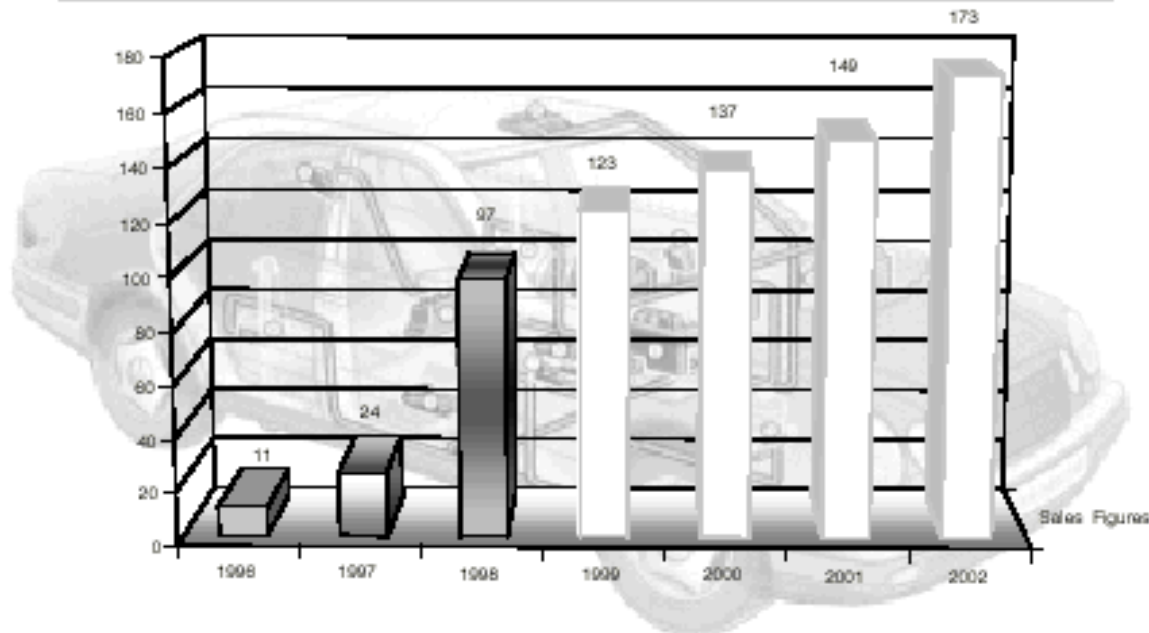
L'**RTR bit** è posto alla fine dell'identificatore esteso (18 bit).

L'**SRR bit** (Substitute Remote Request), che prende il posto dell'RTR bit nell'**EXTENDED FRAME**, è sempre recessivo.

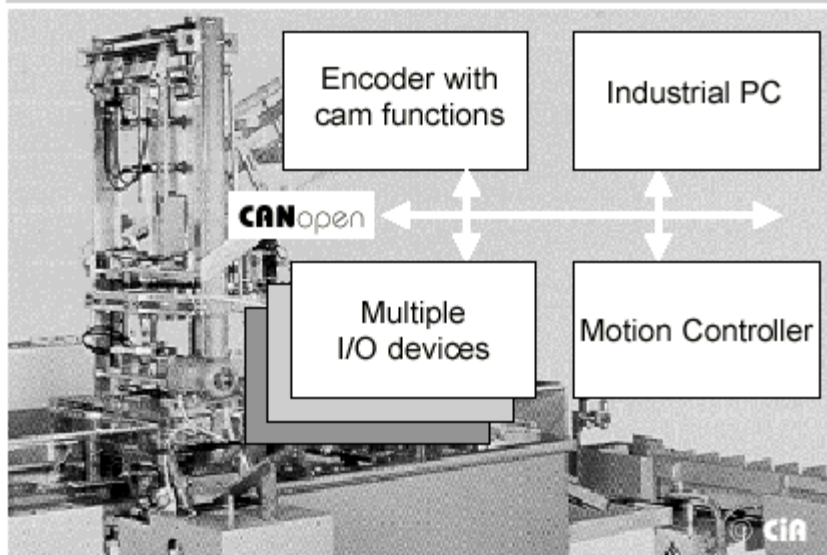
L'**IDE bit** se dominante identifica uno **STANDARD FRAME**, se recessivo un **EXTENDED FRAME**.

In campo **automobilistico**, sono sempre più adottate reti CAN per il coordinamento dei numerosi dispositivi elettronici a bordo di alcune autovetture (Mercedes classe C ed E, Jaguar XK8, BMW e numerose altre);

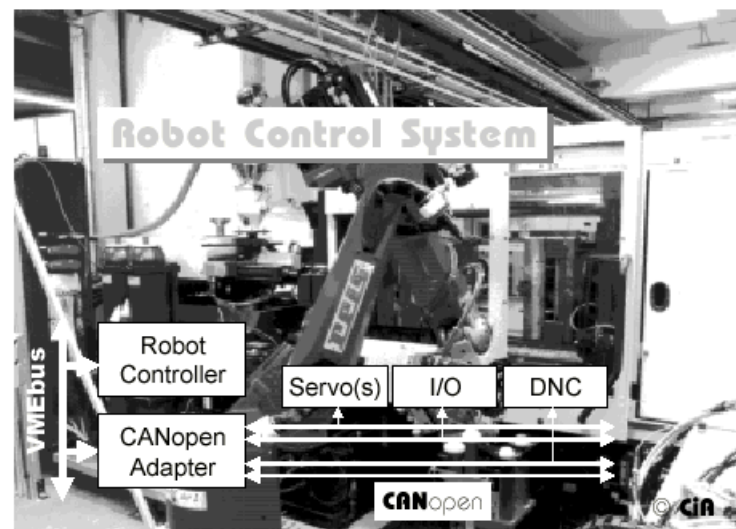
CAN Node Sales Figures



Carton Packaging Machine



In campo industriale, si sta tentando di sfruttare al meglio tutti i servizi offerti dal livello di applicazione (livello 7 della pila ISO/OSI) al fine di accrescere le potenzialità di CAN.



In campo medico, CAN permette di semplificare il cablaggio e la gestione di apparecchiature complesse e sofisticate.

X-Ray Angio-Biplane-System

