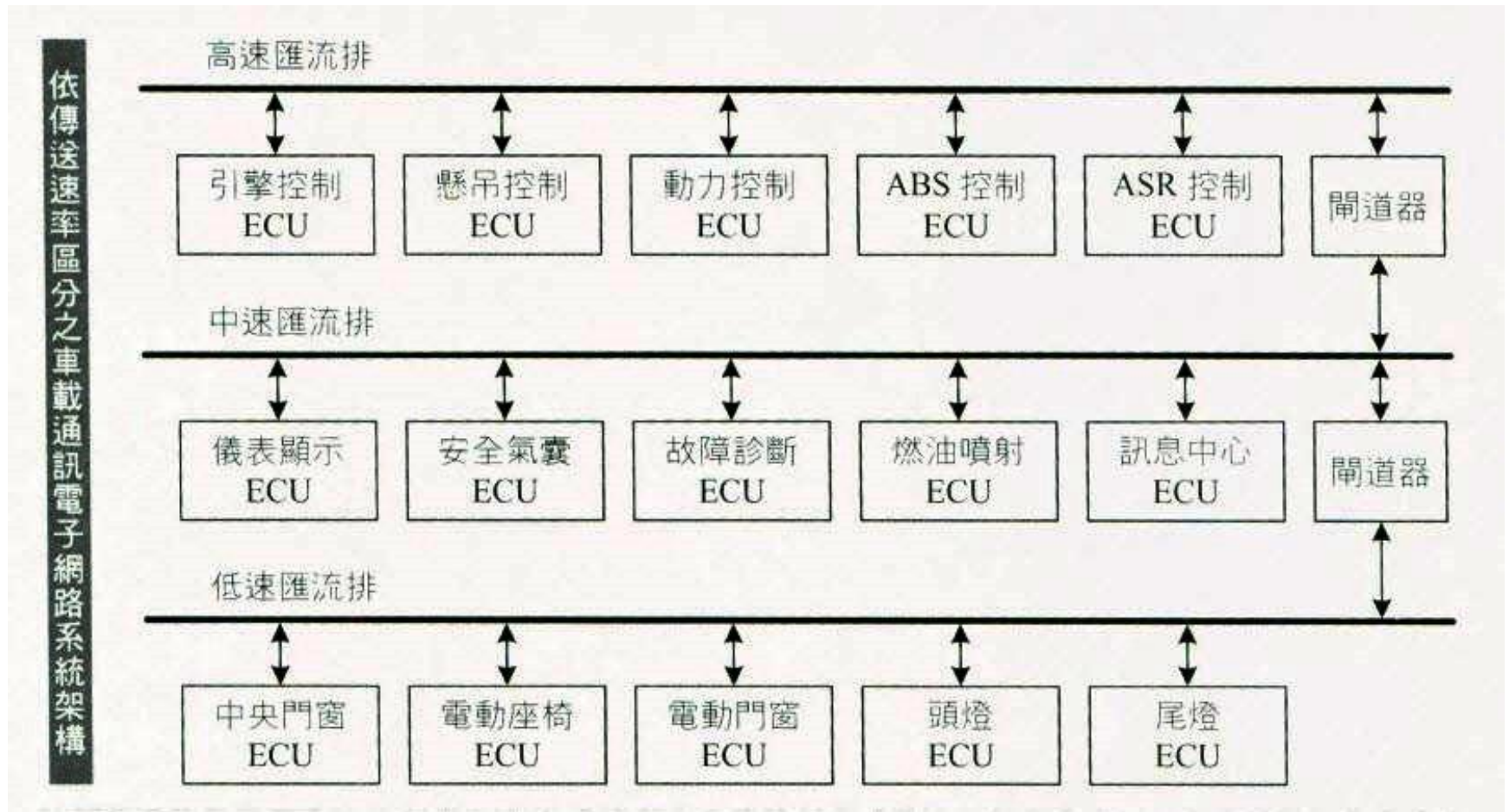


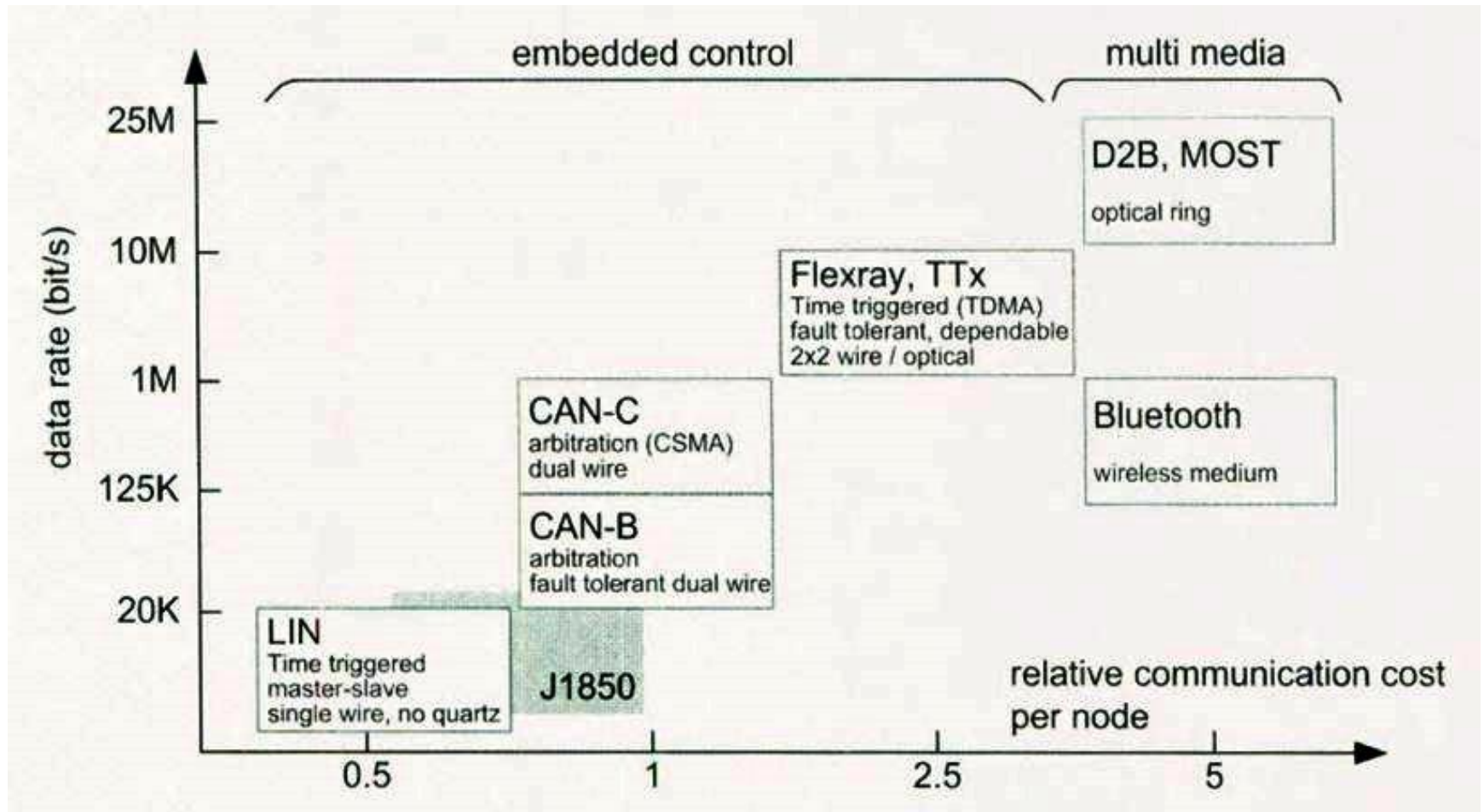
車載網路系統

2012.12.23

車載網路系統架構

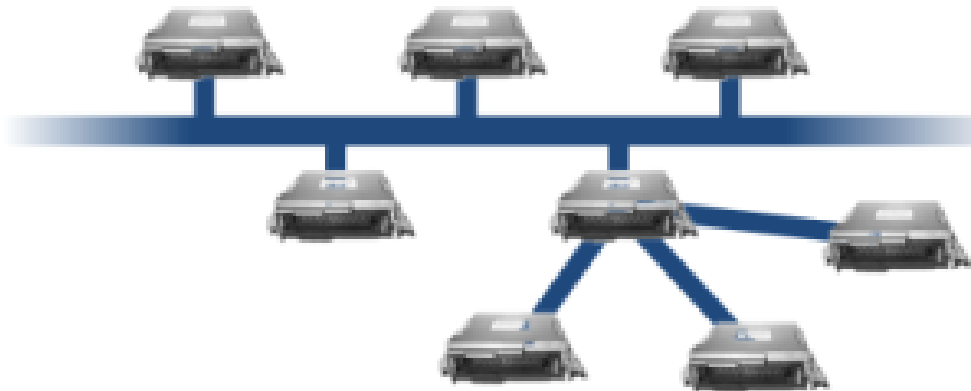


車載網路技術



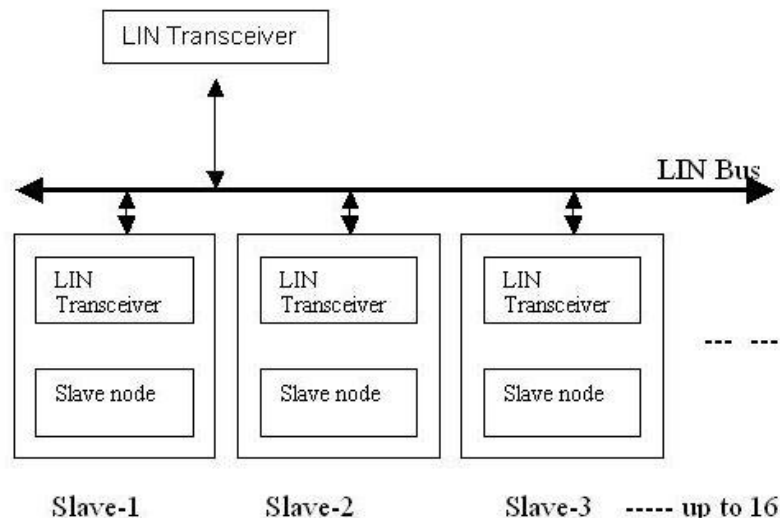
FlexRay

- **FlexRay**主要是滿足二方面的要求：可靠性和速率。
- 可靠性：由二個FlexRay ECU和二條匯流排構成一個網路，每個ECU和二條匯流排連接。
- 速率：使用TDMA (Time Division Multiple Access)的存取技術，將訊息分配到確定的時間區段，各設備按照優先順序占用不同的時隙使用匯流排傳輸資料。
- FlexRay使用星形連接和匯流排連接，高速採用點對點構成星形結構，低速採用匯流排結構。

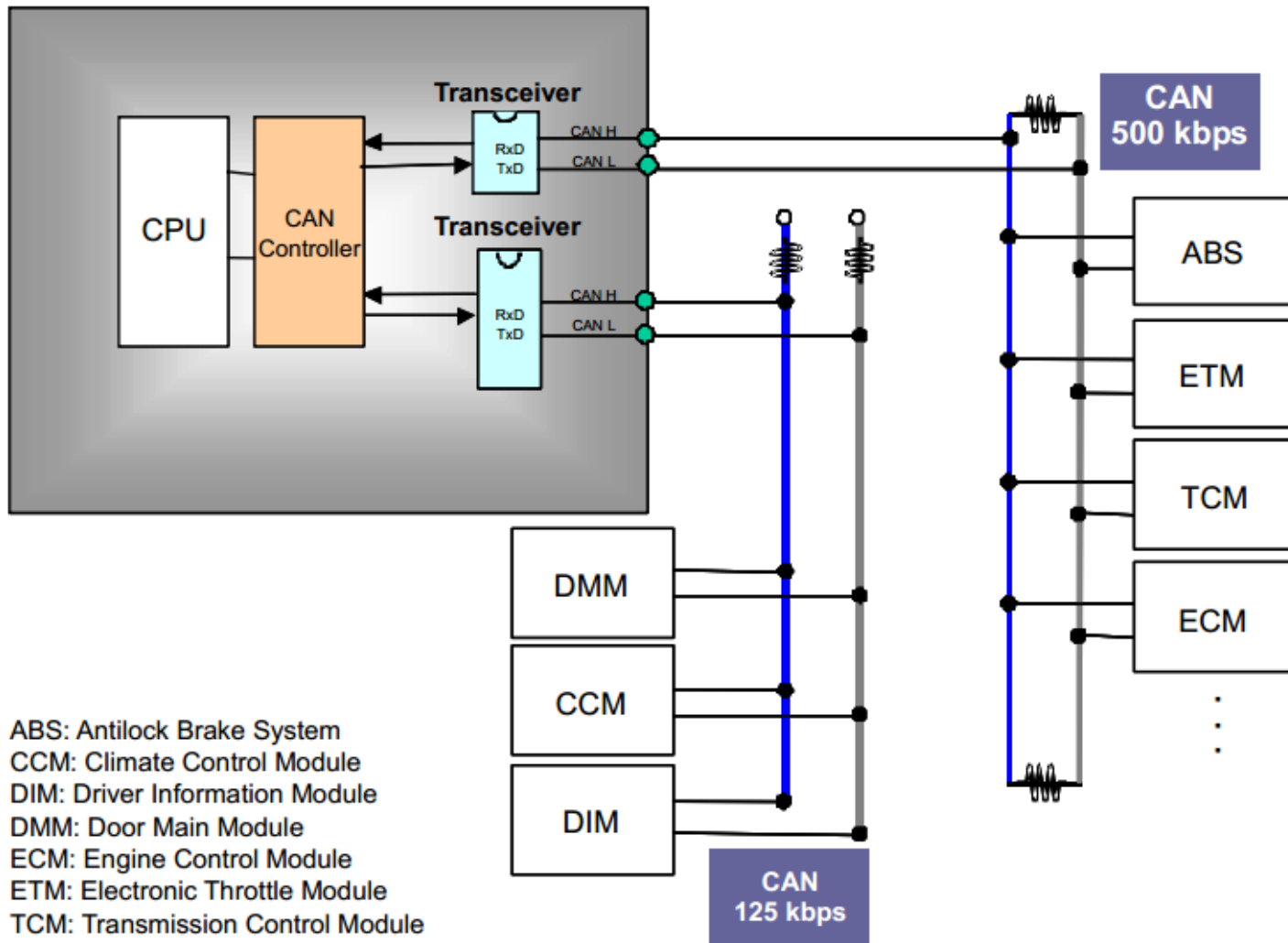


LIN (Local Interconnect Network)

- **LIN**是一種低成本的串列通訊網路協定，採用單主機ECU節點與多組從主機ECU節點的模式。採用12V的信號線，最高傳輸速率20kbps，節點數可以達到16個。
- 透過CAN/LIN閘道器可以和其他系統交換訊息。
- 例如在車門上有門鎖、車窗升降等，只需一個LIN網路。

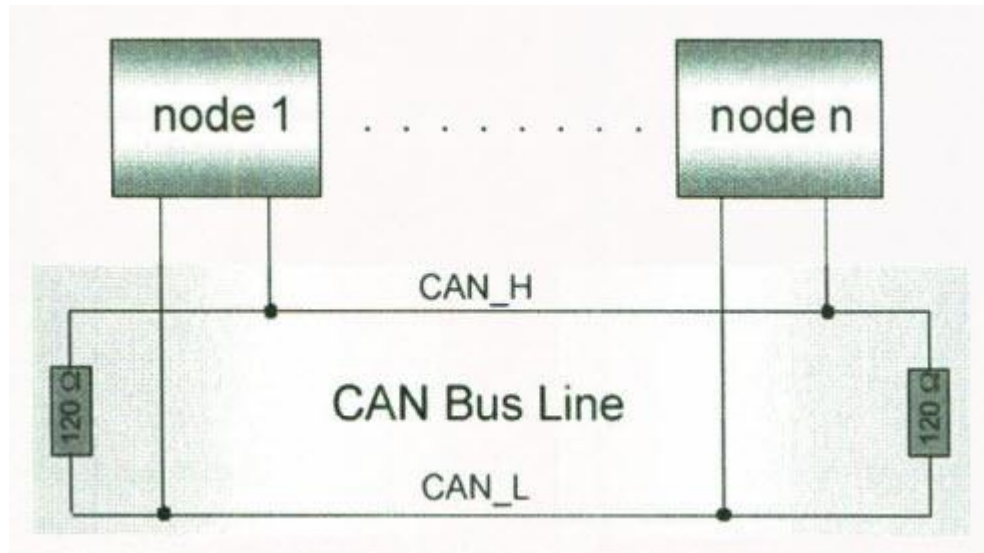


CAN Bus連接圖

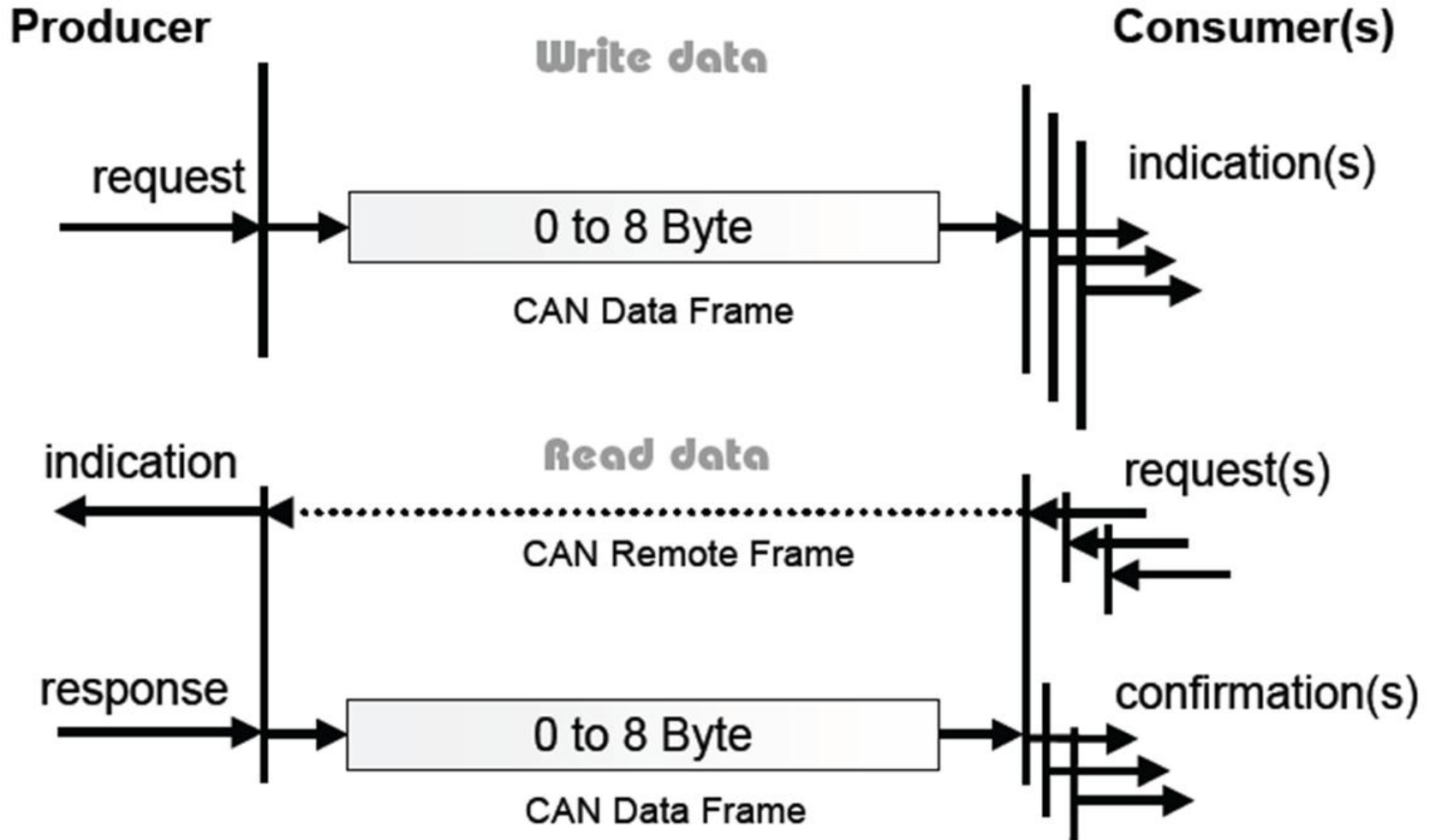


CAN Bus基本概念

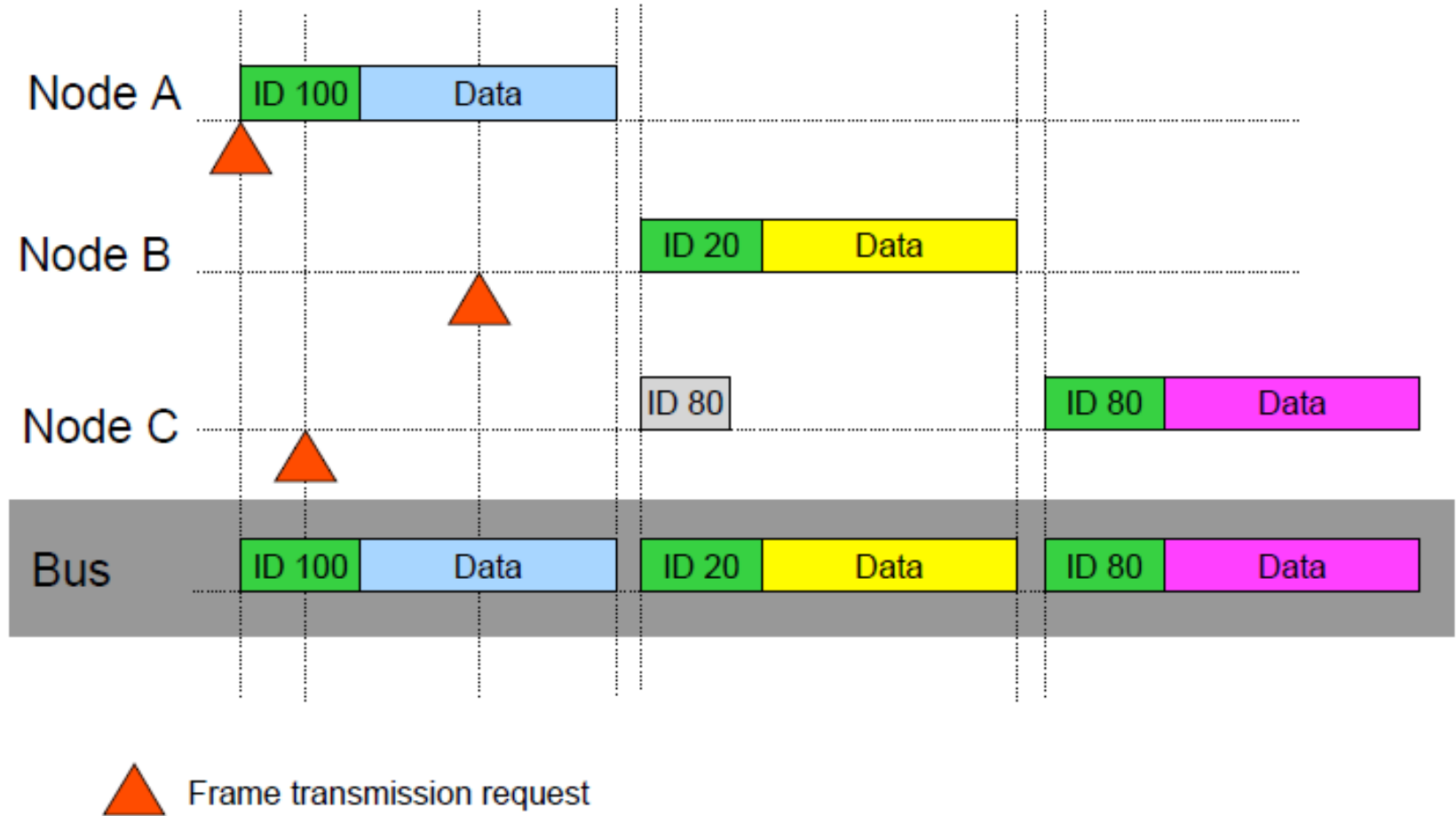
- 汽車上有10-20個電控單元(ECU)，每個ECU與感測器連接，與其他ECU交換資訊都需要通訊網路。
- **CAN** (Controller Area Network)匯流排提供ECU之間的通訊，構成**車載網路系統**。
- 每個ECU都是網路上的一個節點，藉由CAN匯流排的控制器，驅動器，使用二條導線(不含地線)連接到下一個節點。



CAN通訊服務



CAN Bus上的傳輸資料



CAN發展歷史

- 1983: Bosch開始發展
- 1986: 發表CAN protocol
- 1991: Bosch 發表CAN 2.0 specification
- 1992: CiA, (CAN in Automation)成立
- 1992: CAN Application Layer (CAL)公布
- 1993: ISO 11898
- 1994: DeviceNet protocol
- 1995: CANopen protocol by CiA
- 2000: TTCAN

CAN的基本特性

1. 多主控制

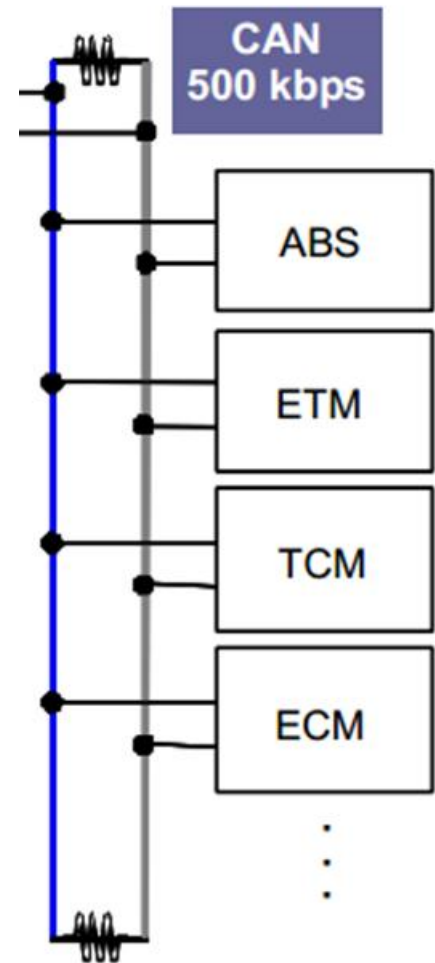
所有節點都可以傳送訊息，採用載波檢測多重存取/碰撞避免(CSMA/CA)，當有多個節點同時傳送，具有高優先權識別碼ID的ECU可優先傳送。ID越小，優先權越高。

2. 訊息的傳送：

所有訊息使用相同格式。當二個信息同時傳送時，根據ID逐一比較位元，ID較大的停止傳送，不會因碰撞造成無法傳送。

3. 系統的相容性：

每個節點只要設定ID，沒有位址的訊息，不需改變軟硬體。



CAN的基本特性

4. 通訊速度：

同一網路的速度相同，不同速度的網路藉由閘道器連接。

5. 遠端資料請求：

藉由遠端訊框請求其他單元傳送資料。

6. 錯誤檢測、通知與恢復功能

所有節點都會進行檢測，若有錯誤會通知其他節點，並結束目前的傳送。發生錯誤的訊息會重複傳送直到成功為止。

7. 故障隔離

持續有傳送或接收錯誤的節點會隔離，直到其恢復正常再加入系統。

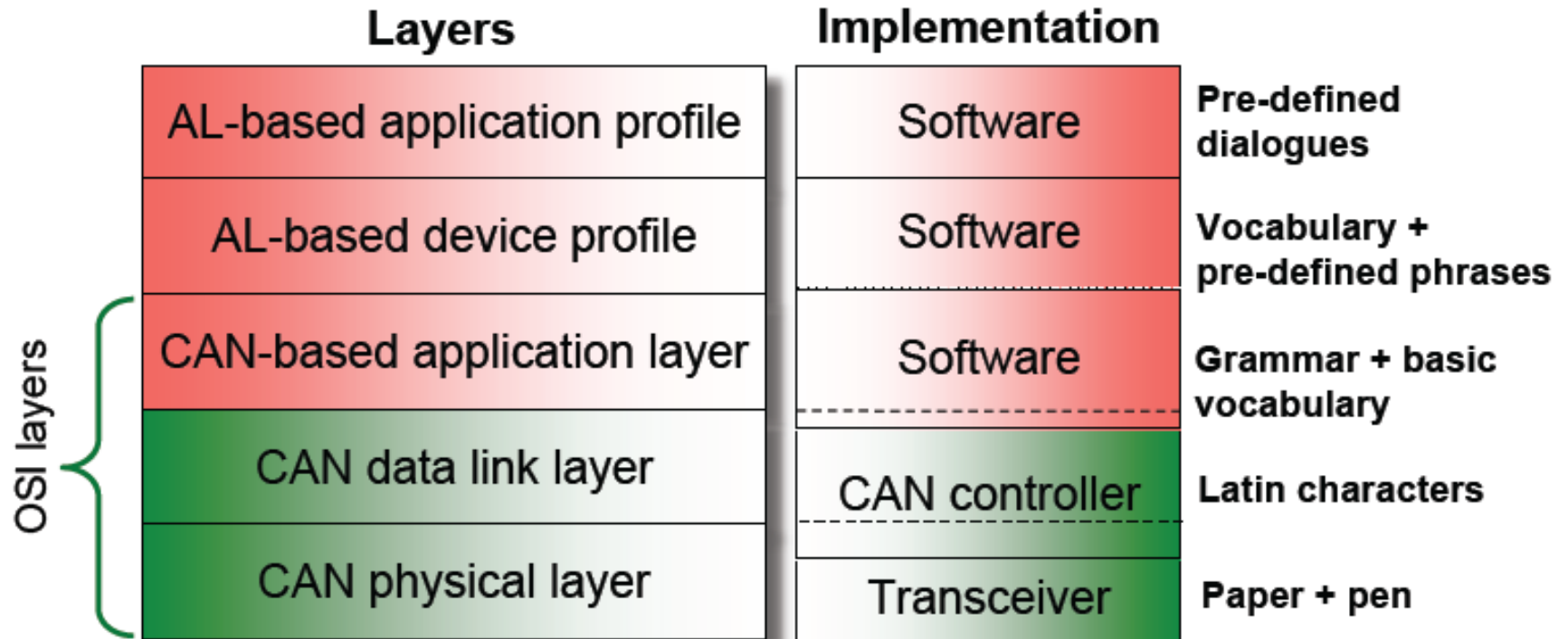
ISO OSI網路模型

ISO/OSI 基本模型		功能
軟體 控制	7 應用層	由實際應用程式提供可利用的服務。
	6 描述層	進行資料表現形式的轉換。例如，文字設定、資料壓縮、加密等的控制。
	5 會談層	建立會談式的通訊，控制資料正確地接收和傳送。
	4 傳輸層	控制資料傳輸的順序、傳送錯誤的恢復等，保證通訊的品質。例如，錯誤修正、再傳輸控制。
	3 網路層	進行資料傳送的路由選擇或中繪。例如，單元間的資料交換、位址管理等。
硬體 控制	2 資料連結層	將實體層收到的信號(位先序列)組成有意義的資料，提供傳輸錯誤控制等資料傳輸控制流程。例如，存取的方法、資料的形式。
	1 實體層	規定使用的電纜、連接器、電氣信號規格等，以實現設備間的信號傳送。

- CAN規範了模型中的最下面二層：**實體層**和**資料連結層**。這二層完全由硬體實現。
- **應用層**由使用者自行定義。汽車工業的製造商各有自己的標準，互不相容。
- 在工業控制和製造業廣泛應用的標準是device net，這是為PLC和智慧型感測器設計的。

ISO/OSI 基本參照模型		CAN 網路模型	功能
軟體 控制	7 應用層	應用層	
	6 描述層		
	5 會談層		
	4 傳輸層		
	3 網路層		
硬體 控制	2 資料連結層	資料連結層	訊息的格式化、仲裁、回應等
	1 實體層	實體層	信號實際的傳送方式

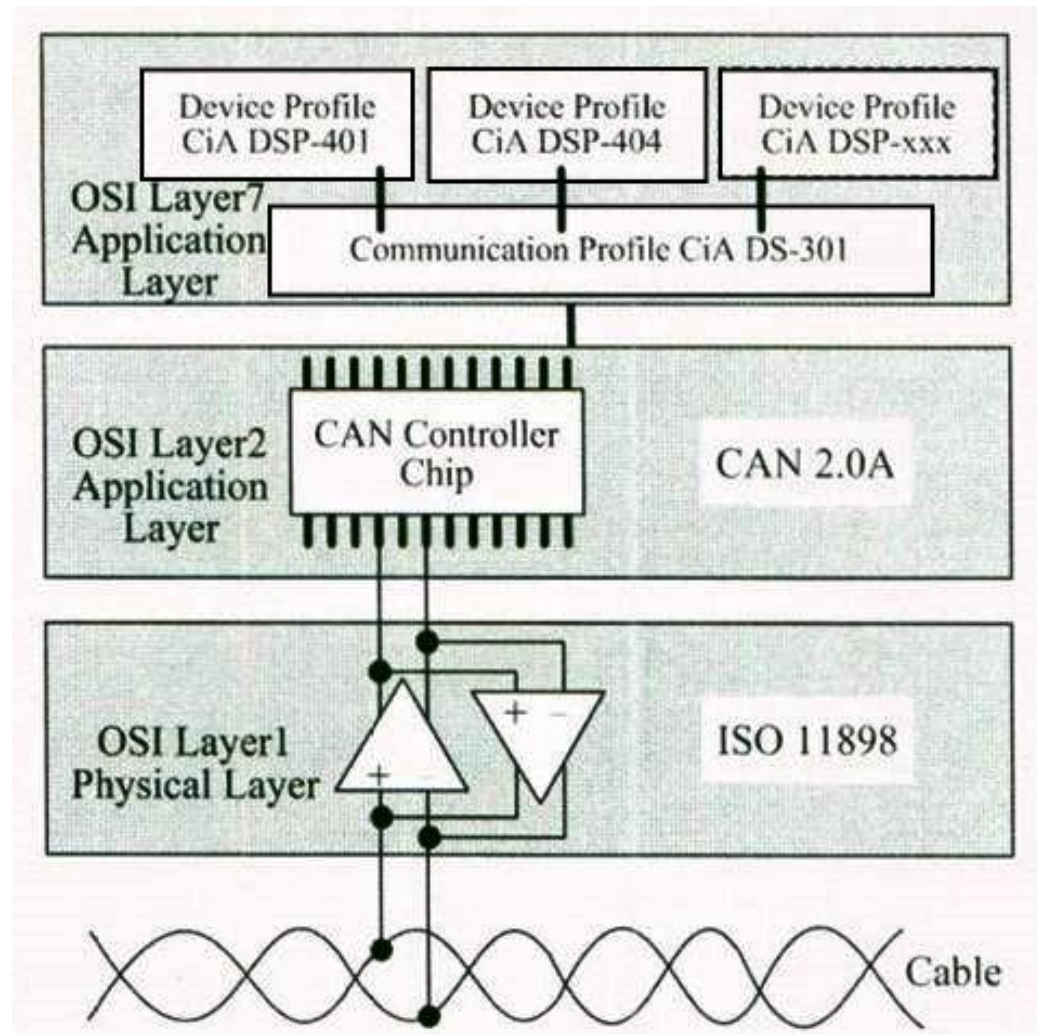
CAN reference model



CAN協定的基本概念

- CAN協定涵蓋ISO規定的OSI模型中的**傳輸層**、**資料連結層**和**實體層**。
- **資料連結層**的功能是將實體層收到的信號組合成有意義的訊息，並提供傳送錯誤控制等傳輸控制的流程。例如訊息的格式化、仲裁、回應、錯誤檢測或報告。
- **資料連結層**分為MAC子層和LLC子層，通常在CAN控制器的硬體執行。MAC子層是CAN協定的核心部分。
- **實體層**定義信號實際的傳送方式、位元時脈、位元編碼、同步的方法。例如信號電位、通訊速度、取樣點、驅動器、匯流排的電氣特性。

CAN協定的標準應用內容



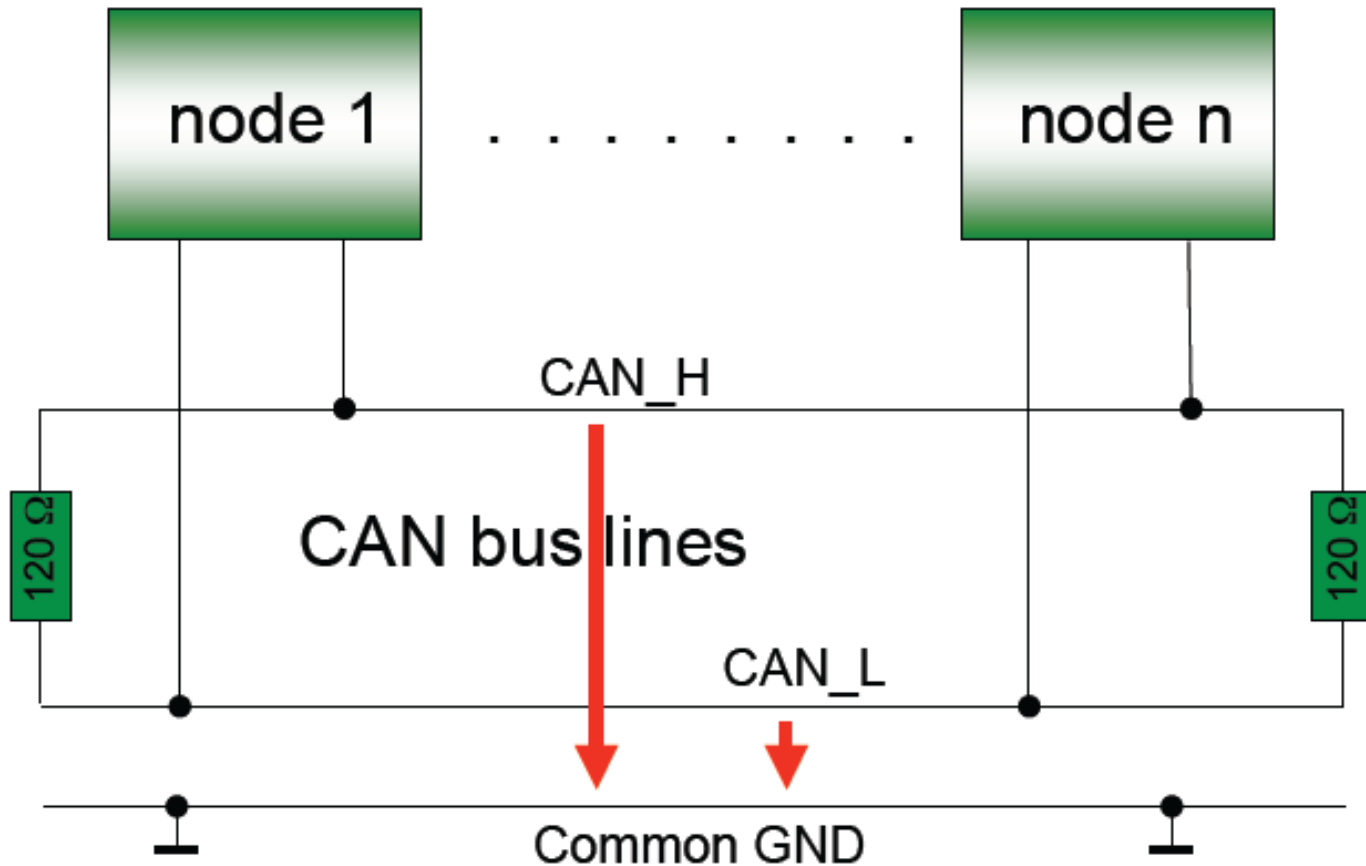
CAN協定及標準規格

應用層		
資料連結層 ISO11898/ISO11519-2相同		資料連結層和部分的實體層在ISO已經標準化
實體層 ISO11898/ISO11519-2 不同	PLS層	
	PMA層	PMA和MDI子層在ISO11898和11519-2不同
	MDI層	
CAN協定在實體層對收發器、連接器、纜線沒有標準化		

Physical layer requirements

- ◆ Non-return-to-zero coding
- ◆ Bit-stuffing
- ◆ In bit-time response
- ◆ Automatic re-synchronization
- ◆ Dominant and recessive bit representation

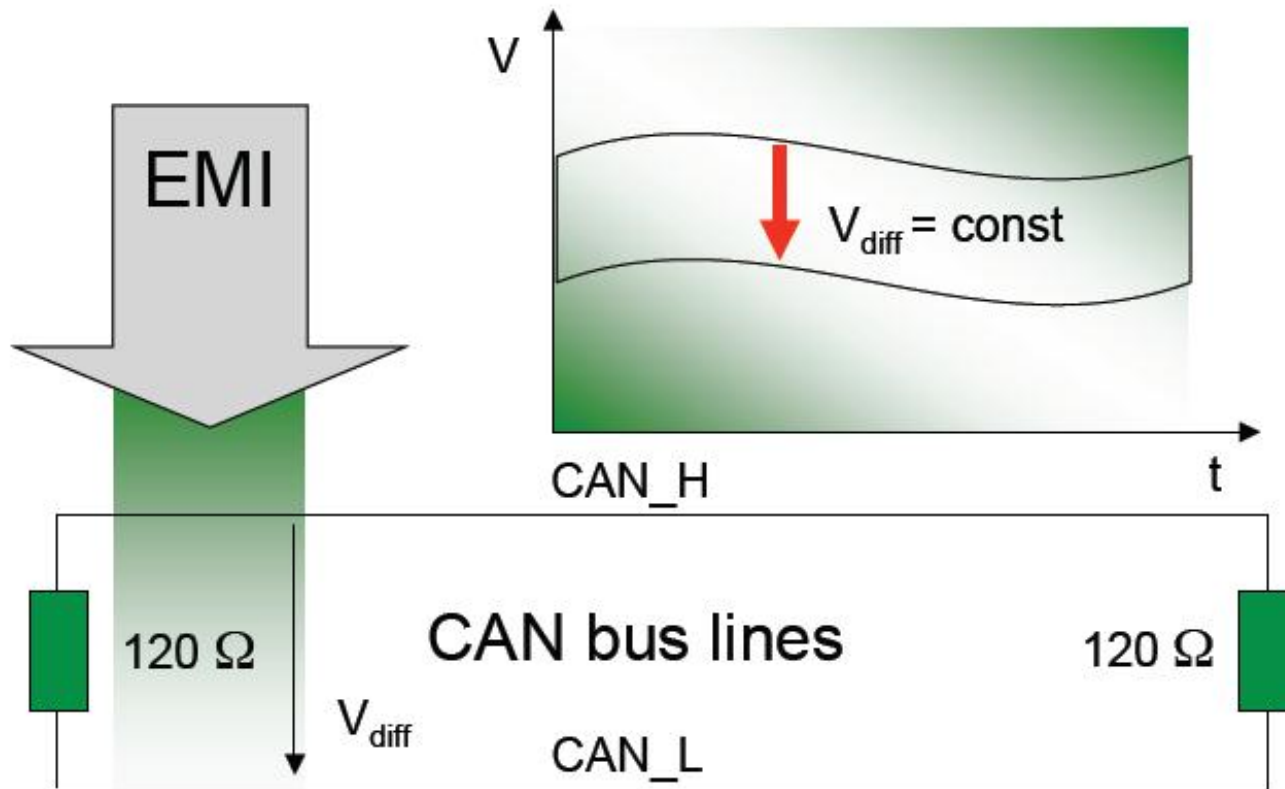
CAN bus ISO 11898-2



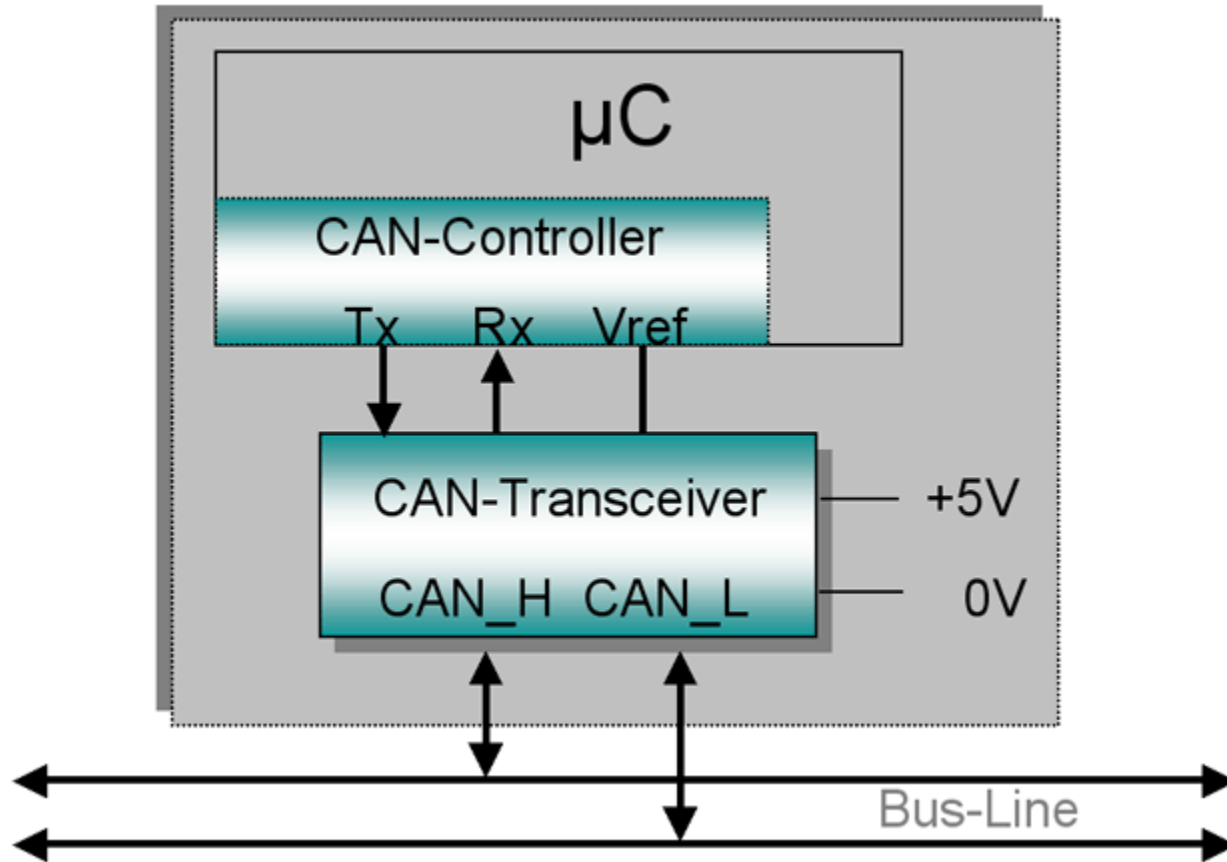
- 匯流排二端要安裝終端電阻。

CAN bus 電磁干擾

- 匯流排線路受到的干擾相同，使用差分輸入可以消除干擾的影響。



CAN bus ISO 11898-2 node



- Vref的輸出電壓為 $V_{cc}/2$.

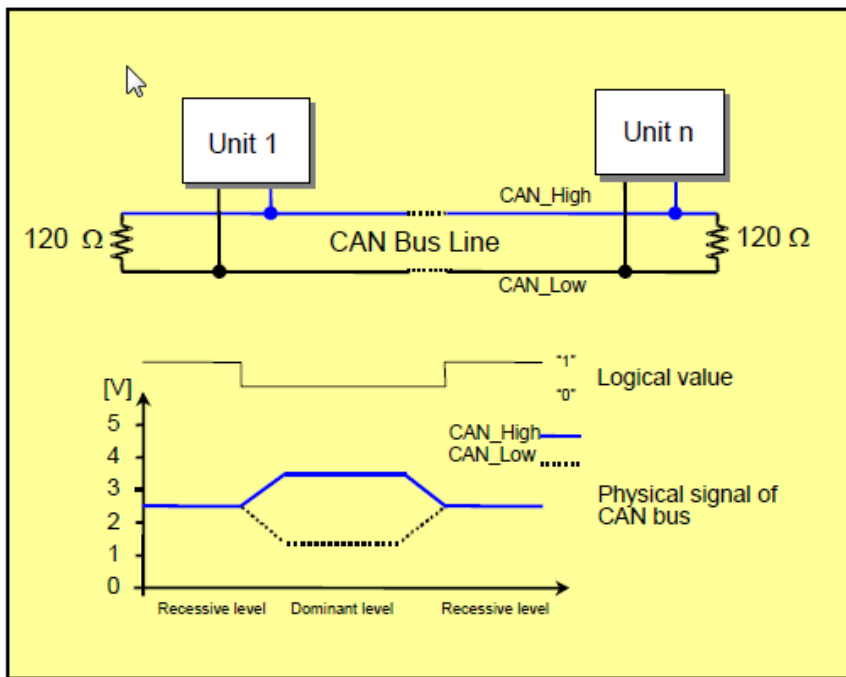
CAN bus Non-Return-to-Zero-Coding

- 每一個位元的大小區分為 'dominant' 或 'recessive' .
- 在每一個位元不一定出現下降或上升邊緣，可能由於振盪器的偏移，以致失去同步。

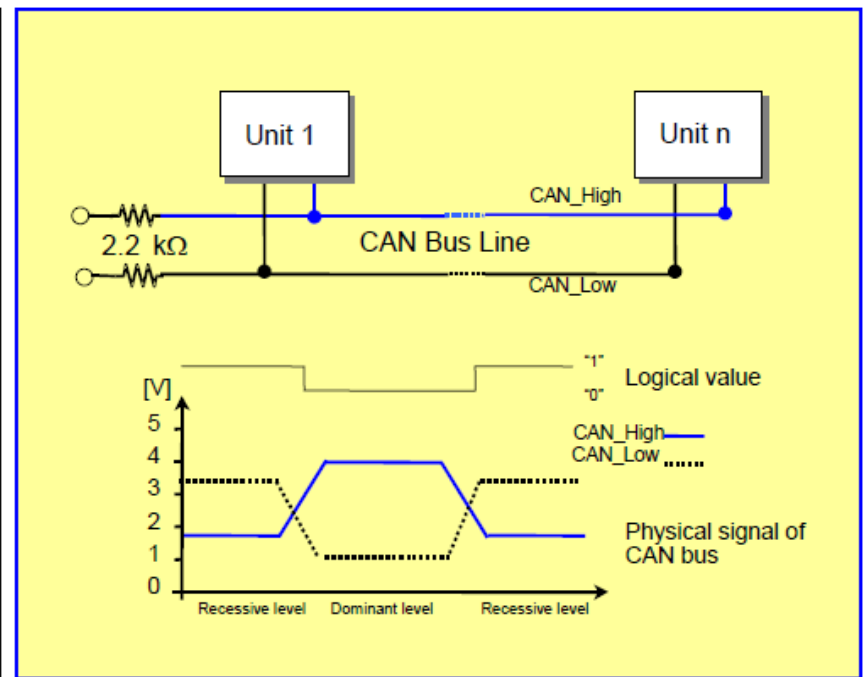


ISO11898和ISO11519-2的實體層

- CAN收發器根據二條匯流排引線(CAN_H)和(CAN_L)的電位差來判斷匯流排電位。
- 匯流排電位分為顯性Dominant(0)和隱性Recessive(1)二種。



[For ISO11898 (125 kbps to 1 Mbps)]



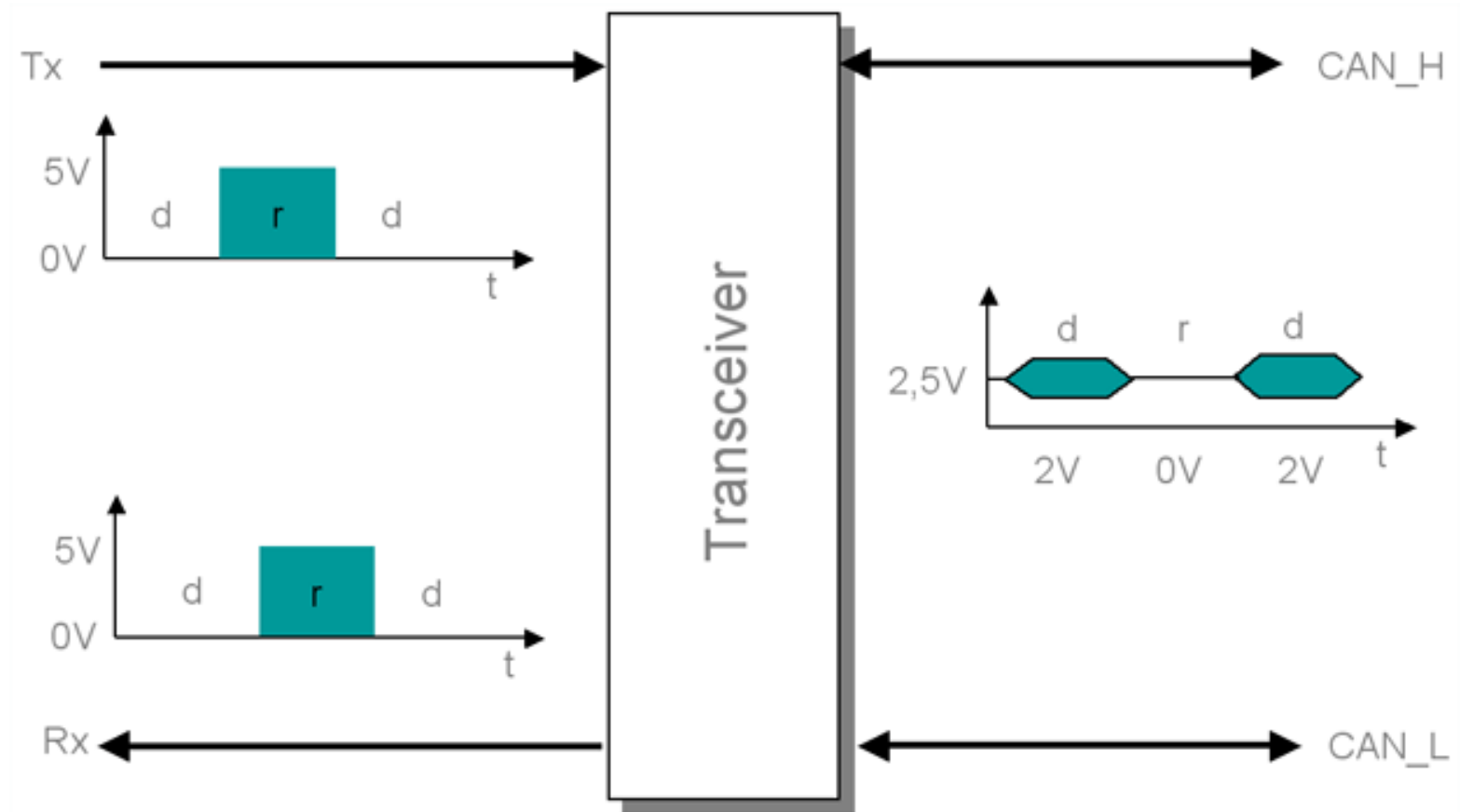
[For ISO11519-2 (10 kbps to 125 kbps)]

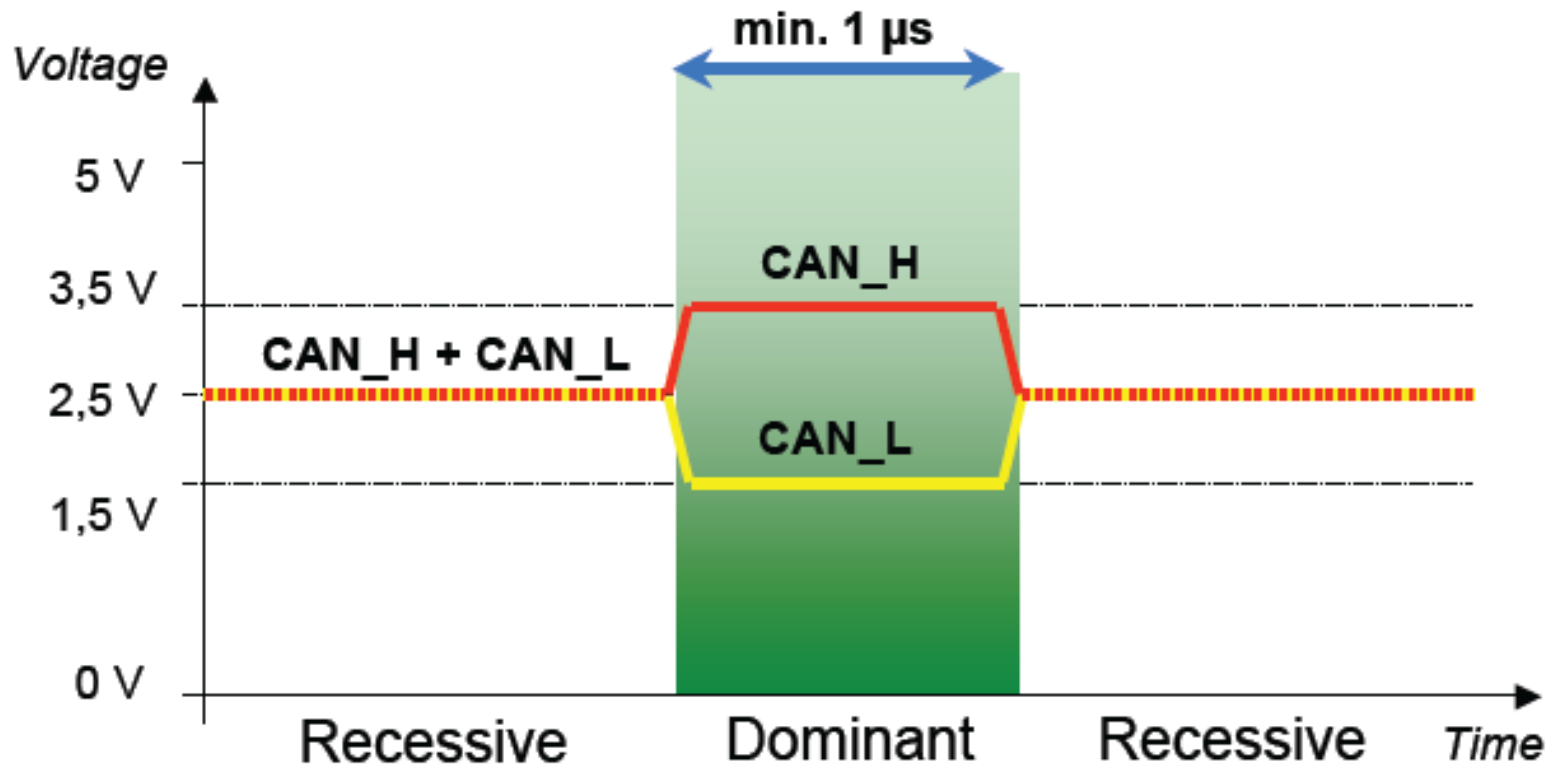
ISO 11898和11519-2實體層的主要差異

Physical layer	ISO 11898 (High speed)						ISO 11519-2 (Low speed)					
Communication speed	Up to 1 Mbps						Up to 125 kbps					
Maximum bus length	40 m/ 1 Mbps						1 km/ 40 kbps					
Number of connected units	Maximum 30						Maximum 20					
Bus topology ^{*1}	Recessive			Dominant			Recessive			Dominant		
	Min.	Nom.	Max.	Min.	Nom.	Max.	Min.	Nom.	Max.	Min.	Nom.	Max.
CAN_High (V)	2.00	2.50	3.00	2.75	3.50	4.50	1.60	1.75	1.90	3.85	4.00	5.00
CAN_Low (V)	2.00	2.50	3.00	0.50	1.50	2.25	3.10	3.25	3.40	0.00	1.00	1.15
Potential difference (H-L) (V)	-0.5	0	0.05	1.5	2.0	3.0	-0.3	-1.5	-	0.3	3.00	-
	Twisted pair wire (shielded/unshielded) Loop bus Impedance (Z): 120 Ω (Min. 85 Ω, Max. 130 Ω) Bus resistivity (r): 70 MΩ/m Bus delay time: 5 ns/m Terminating resistance: 120 Ω (Min. 85 Ω, Max. 130 Ω)						Twisted pair wire (shielded/unshielded) Open bus Impedance (Z): 120 Ω (Min. 85 Ω, Max. 130 Ω) Bus resistivity (r): 90 MΩ/m Bus delay time: 5 ns/m Terminating resistance: 2.20 Ω (Min. 2.09 Ω, Max. 2.31 Ω) CAN_L and GND capacitance: 30 pF/m CAN_H and GND capacitance: 30 pF/m CAN_L and GND capacitance: 30 pF/m					

CAN bus ISO 高速傳輸信號

- TX, RX為5V，經過Transceiver轉換為CAN_H和CAN_L





recessive: 如果CAN_H的電壓不高於CAN_L的電壓 + 0.5 V.

Dominant: 如果CAN_H的電壓至少大於CAN_L的電壓0.9V.

在dominant 狀態，CAN_H 的電壓為3.5 V，CAN_L的電壓為1.5V.

收發器IC的選擇

- ISO 11898與ISO 11519-2實體層規格不同，對應的收發器IC也需要相對應。

Transceiver IC	
ISO11898	ISO11519-2
HA13721RPJE(RENESAS)	TJA1054T(Philips)
TJA1050T(Philips)	TLE6254-3G(Infineon)
TLE6250G(Infineon)	
CF150C(BOSCH)	

CAN standardization

Application Profile		CANopen application profiles	SAE J1939-based Application Profiles
Device Profile	CIP device profiles	CANopen device profiles	e.g. J1939-71, ISO 11783, NEMA2000
Application Layer	DeviceNet IEC 62026	CANopen EN 50325-4	
Data Link Layer	ISO 11898-1 (11-bit ID)	ISO 11898-1 (11-bit and 29-bit ID)	ISO 11898-1 (29-bit ID)
Physical Layer	ISO 11898-2		

CAN協定與標準規格

Name	Baud rate	Specification	Application field
SAE J1939-11	250 k	Two-wire shielded twisted pair	Truck, bus
SAE J1939-12	250 k	Two-wire shielded twisted pair 12 V supply	Agricultural machine
SAE J2284	500 k	Two-wire twisted pair (non-shielded)	Automobile (high-speed: power train system)
SAE J2411	33.3 k, 83.3 k	One-wire	Automobile (low-speed: body system)
NMEA-2000	62.5 k, 125 k, 250 k, 500 k, 1 M	Two-wire shielded twisted pair Power supply	Ship
DeviceNet	125 k, 250 k, 500 k	Two-wire shielded twisted pair 24 V supply	Industrial equipment
CANopen	10 k, 20 k, 25 k, 50 k, 125 k, 250 k, 500 k, 800 k, 1 M	Two-wire twisted pair Optional (shielded, power supply)	Industrial equipment
SDS	125 k, 250 k, 500 k, 1 M	Two-wire shielded twisted pair Optional (power supply)	Industrial equipment

通訊協定分類一覽表

		車載通訊 電子網路類別	通訊速度	用途	協定
電 通 信	低速 CAN (~125kbps)	A 類	~10kbps (車身系統)	燈光類、電動窗、門鎖、 電動椅、遠端門鎖等	<ul style="list-style-type: none"> 各汽車廠商 自有協定 LIN
		B 類	10kbps~125kbps (狀態資訊系統)	電子儀表、駕駛訊息、 自動空調、故障診斷	<ul style="list-style-type: none"> CAN J1850 VAN
光 通 信	高速 CAN (125kbps~ 1Mbps)	C 類	125kbps~1Mbps (即時控制系統)	引擎控制、變速器控 制、煞車控制、懸吊控 制、ABS 等	<ul style="list-style-type: none"> FlexRay Safe-by-Wire
		D 類	5Mbps~ (多媒體)		<ul style="list-style-type: none"> D2B Optical MOST IEEE1394

【註】類別：SAE 的分類名稱

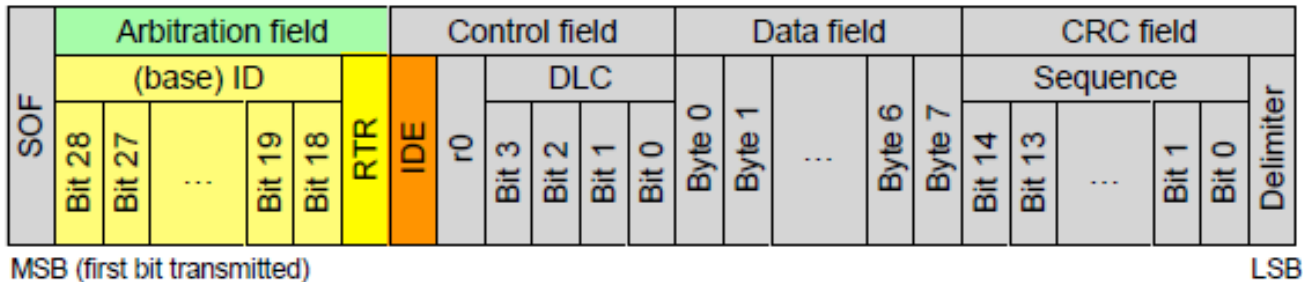
CAN通訊格式類型

- CAN通訊的訊框(frame)類型：
- 資料通訊格式：用於傳送資料
- 遠端通訊格式：請求遠端節點傳送資料
- 錯誤通訊格式：通知其他節點錯誤的格式
- 過載通訊格式：通知其他節點延遲傳送資料的格式
- 通訊格式間隔：與前一個資料間隔的格式

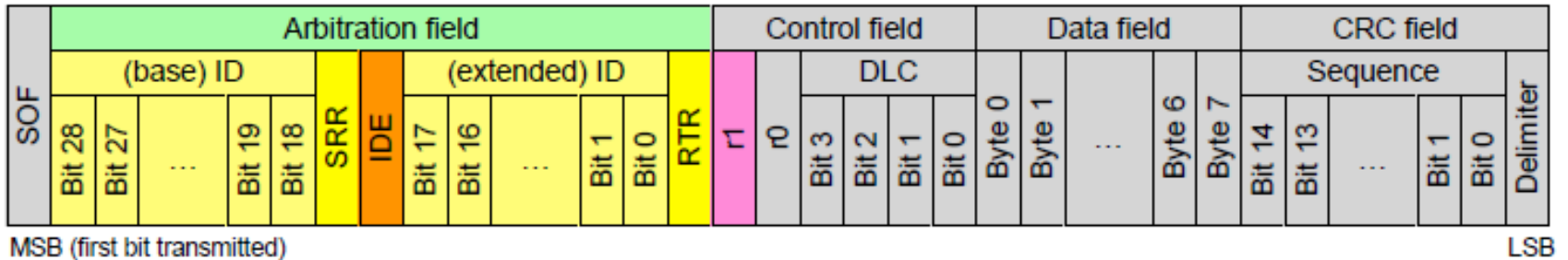
CAN2.0A和2.0B

- CAN2.0A的識別碼為11bits，CAN2.0B的識別碼為29bits。

Base CAN data frame format

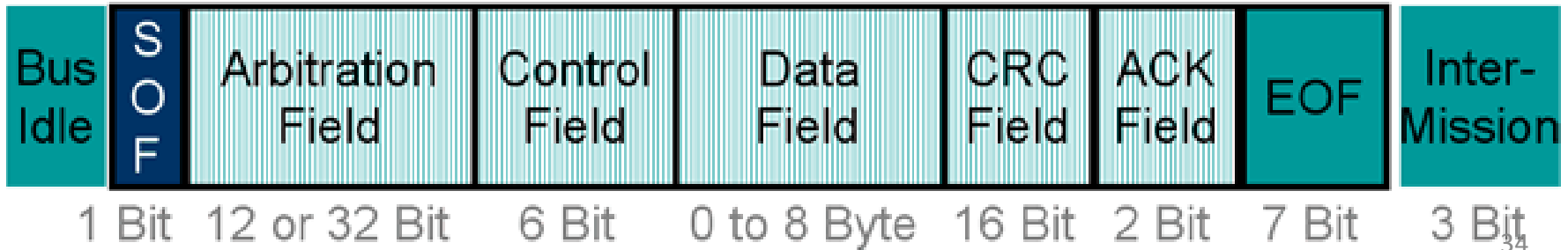


Extended CAN data frame format



資料通訊格式

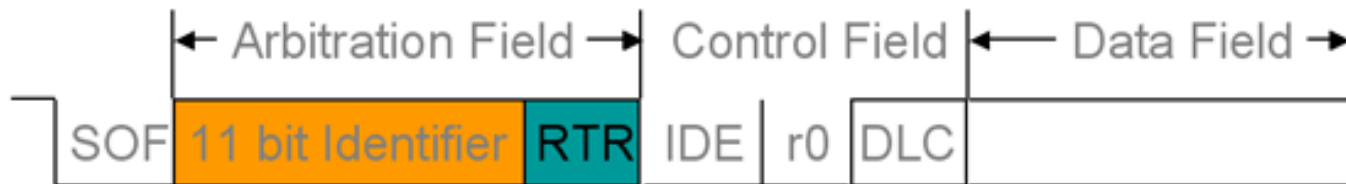
- 起始欄位SOF：送出1位元dominant(0)，用來同步
- 識別欄位：表示節點的優先權，用來仲裁
- 控制欄位：資料的長度和其他控制項目
- 資料欄位：0-8bytes的資料
- CRC：錯誤檢查
- ACK：接收正確送出1位元dominant(0)
- EOF：通訊格式結束的欄位



資料通訊格式-SOF欄位

- 為1位元的dominant(0)欄位，表示frame的開始
- Hard synchronization is performed whenever there is a recessive-to-dominant edge

Standard Frame Format

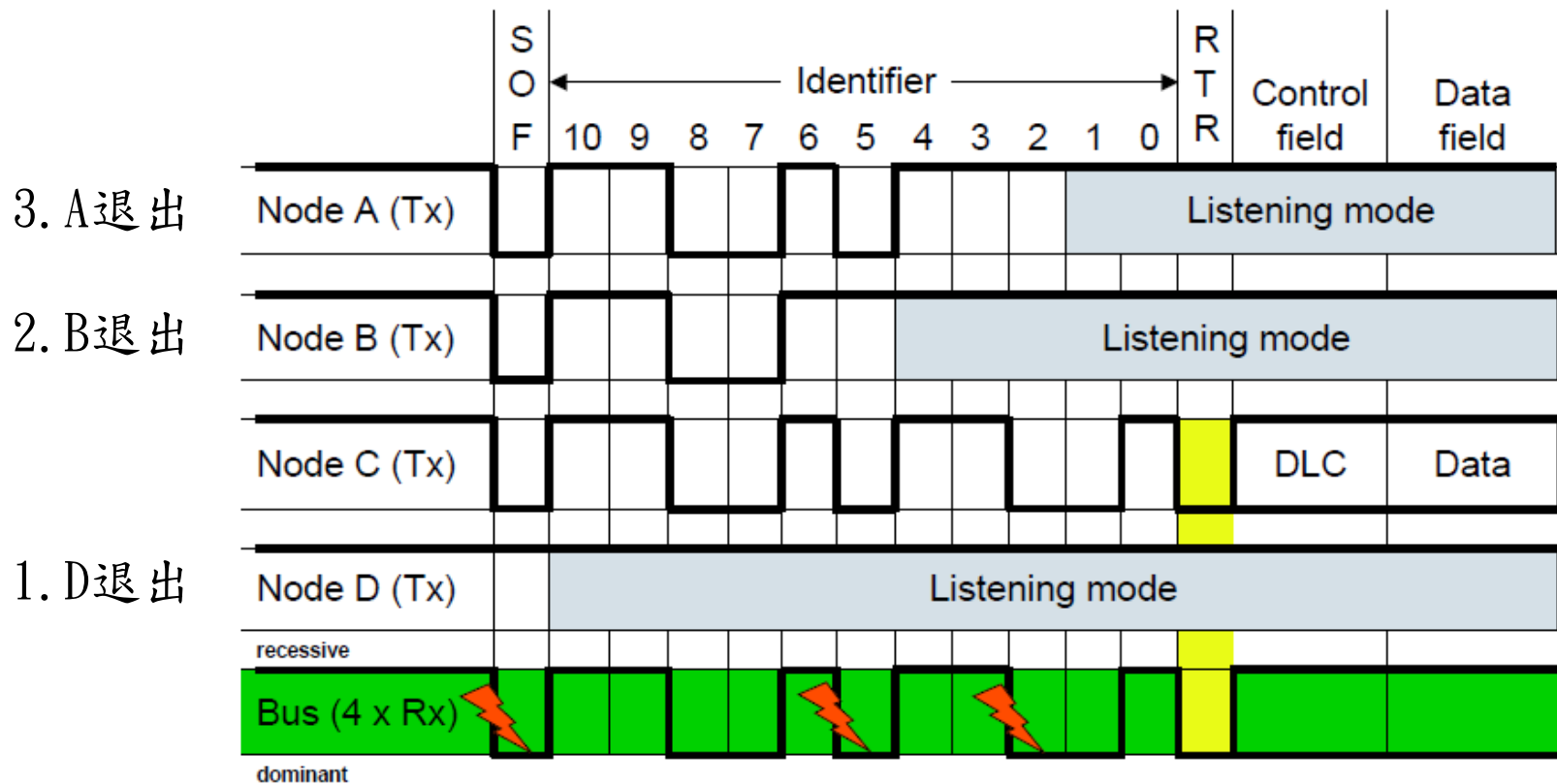


Extended Frame Format



資料通訊格式-仲裁欄位

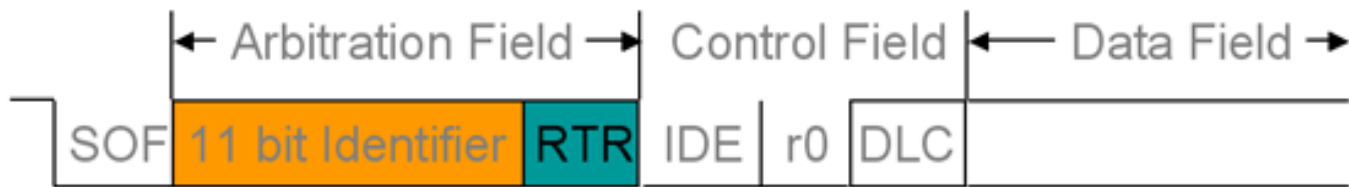
- 表示節點的優先權，ID值越小，優先權越高



資料通訊格式-仲裁欄位

- **RTR bit**:資料通訊格式為**dominant(0)**;遠端資料通訊格式為**recessive(1)**.

Standard Frame Format



Extended Frame Format



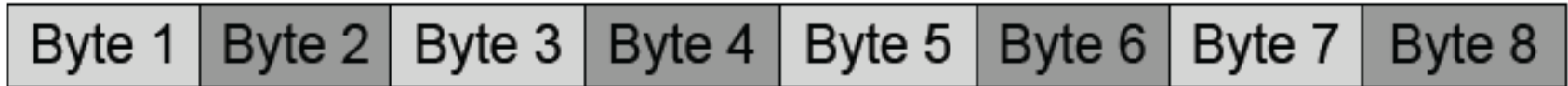
資料通訊格式-控制欄位

RTR	IDE/r1	r0	DLC3	DLC2	DLC1	DLC0	Data/CRC
-----	--------	----	------	------	------	------	----------

- IDE: 標準格式為 dominant(0); 延伸格式為 recessive(1)。
- R0: 保留
- DLC: 資料長度

Data Len.	DLC3	DLC2	DLC1	DLC0
0	d	d	d	d
1	d	d	d	r
2	d	d	r	d
3	d	d	r	r
4	d	r	d	d
5	d	r	d	r
6	d	r	r	d
7	d	r	r	r
8	r	d/r	d/r	d/r

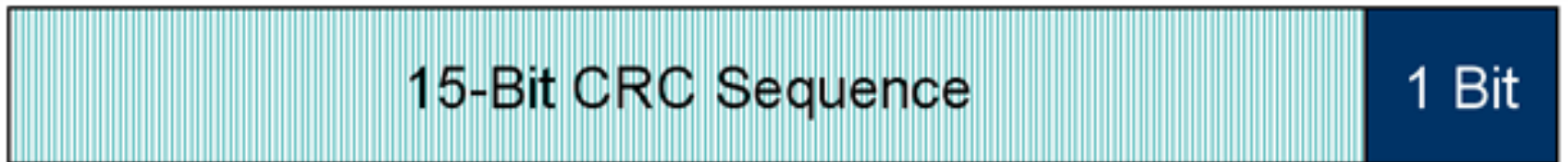
資料通訊格式-資料欄位



- 資料長度0-8 bytes
- 最小資料長度 = 0 Byte
- 最大資料長度 = 8 Byte

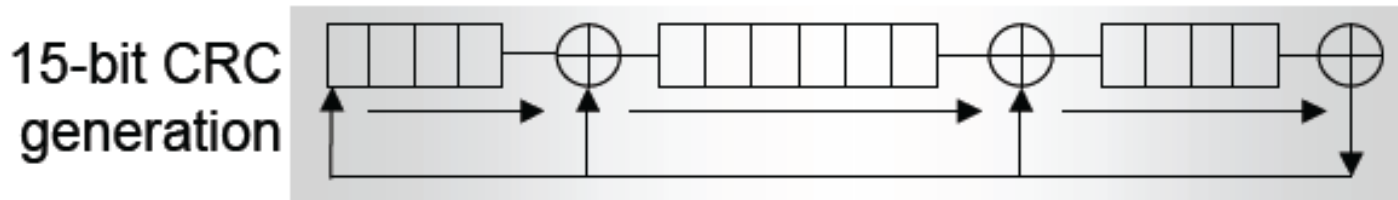
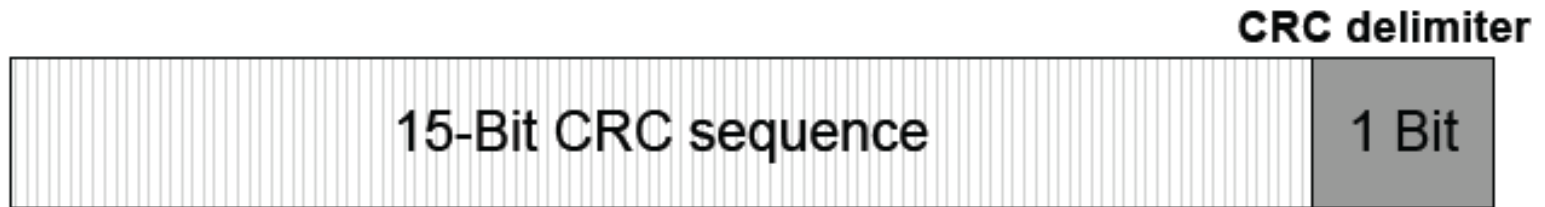
資料通訊格式-CRC欄位

- CRC碼提供的Hamming distance為6,也就是可偵測出5個隨機分布的位元錯誤,其範圍在SOF,仲裁,控制和資料欄位.
- CRC使用的除數為 $x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$



CRC
Delimiter

資料通訊格式-CRC欄位



\oplus Modulo-2-addition

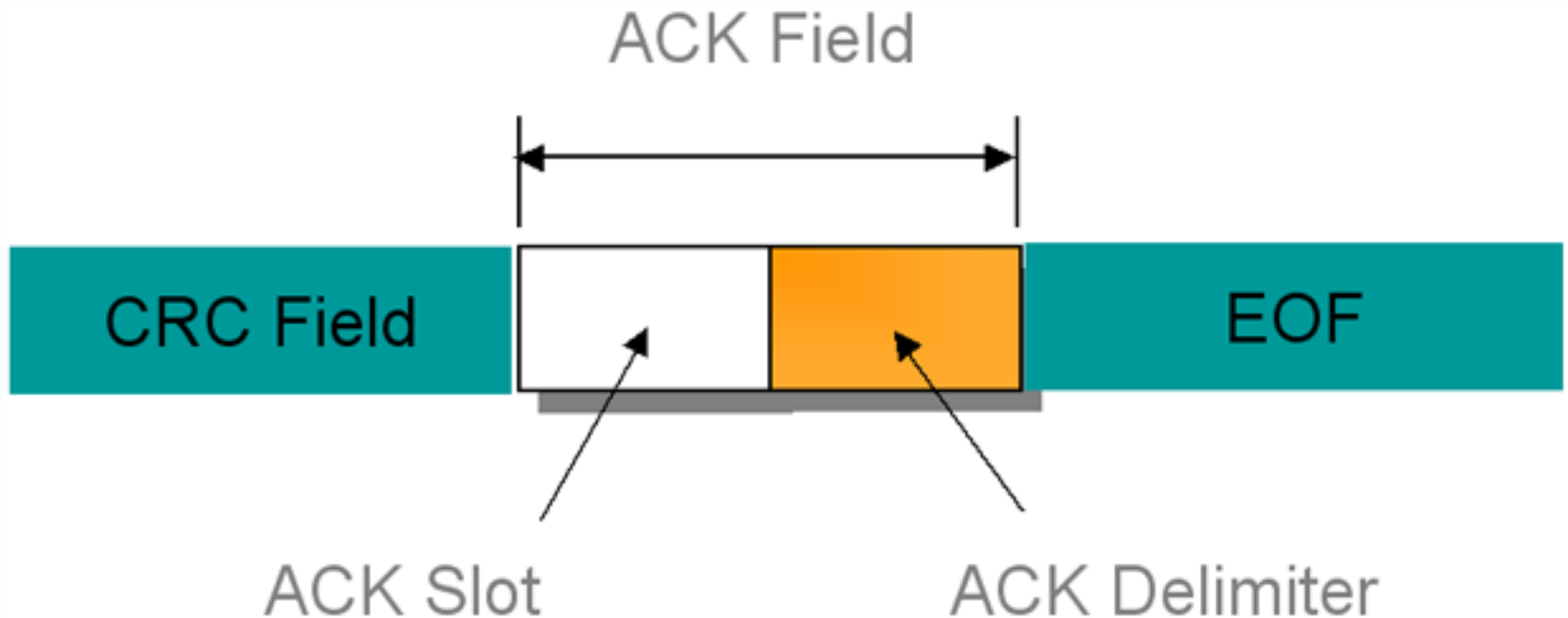
 Shifting register

$$X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1$$

Remark: The CRC delimiter is a fixed formatted recessive bit.

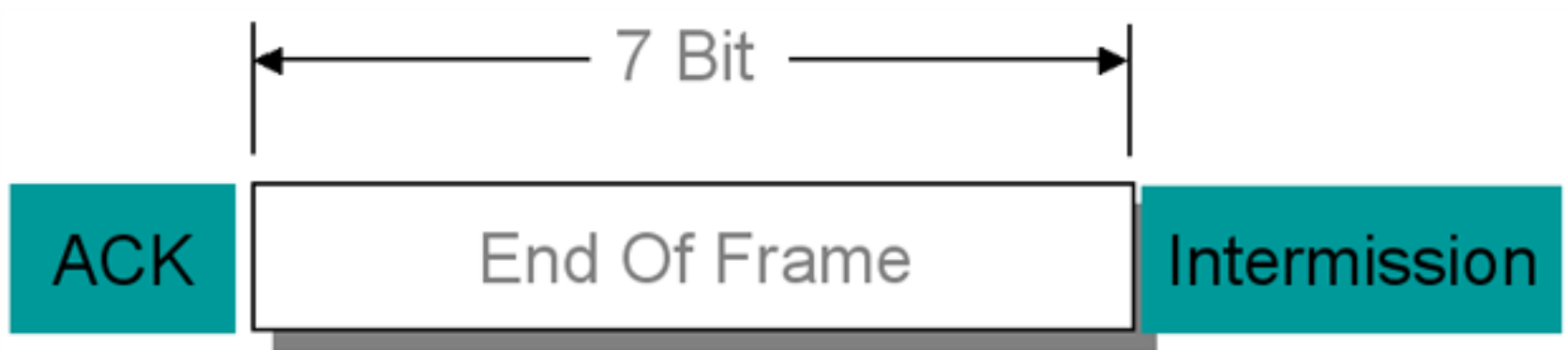
資料通訊格式-確認欄位

- 傳送端送出一個recessive bit(1)
- 當接收端正確收到一個訊息，則在ACK Slot送出一個 **dominant (0)** 位元，告知傳送端.



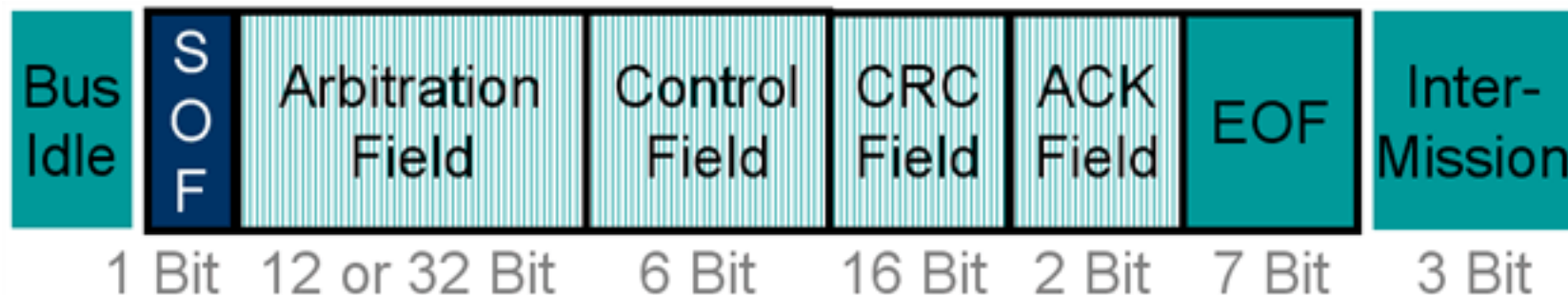
資料通訊格式-結束欄位(EOF)

- 每一個資料和遠端通訊都是由7個連續的recessive(1)位元作為結束。



CAN bus遠端通訊格式

- 當目的節點需要來源節點送出資料時，可藉由送出一個遠端通訊格式來完成，將來源節點的ID寫入仲裁欄位，在RTR填入recessive(1)。

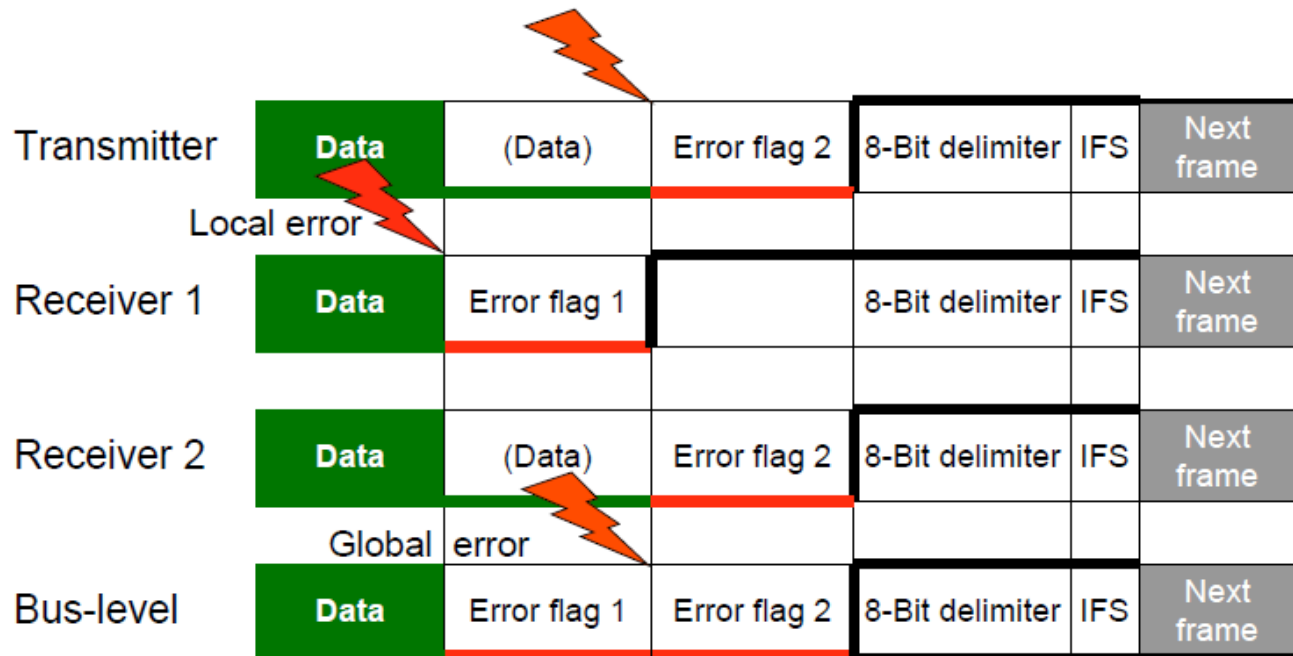


CAN bus 的錯誤處理

1. 偵測Local或global錯誤.
 2. 傳送Error Flag (global錯誤).
 3. 如果是Local錯誤，在Error Delimiter後會跟隨著overlapping Error Flag.
 4. 每一個節點丟棄此訊息.
 5. 每一個節點增加錯誤計數器的值.
 6. 自動重新傳送此訊息.
- 當錯誤發生時，送出6個相同的Error Flag，這會促成bit stuffing error.

CAN bus錯誤處理-Local Errors的全面化

- Receiver1發生local error，傳送6個相同bits的error flag。
- 其他Receiver發現stuffing錯誤，傳送6bits error flag。
- 經過8bits delimiter，3bits interframe。
- 重傳。



Error detection analysis

Probability of non-detected
faulty CAN basic data frames:

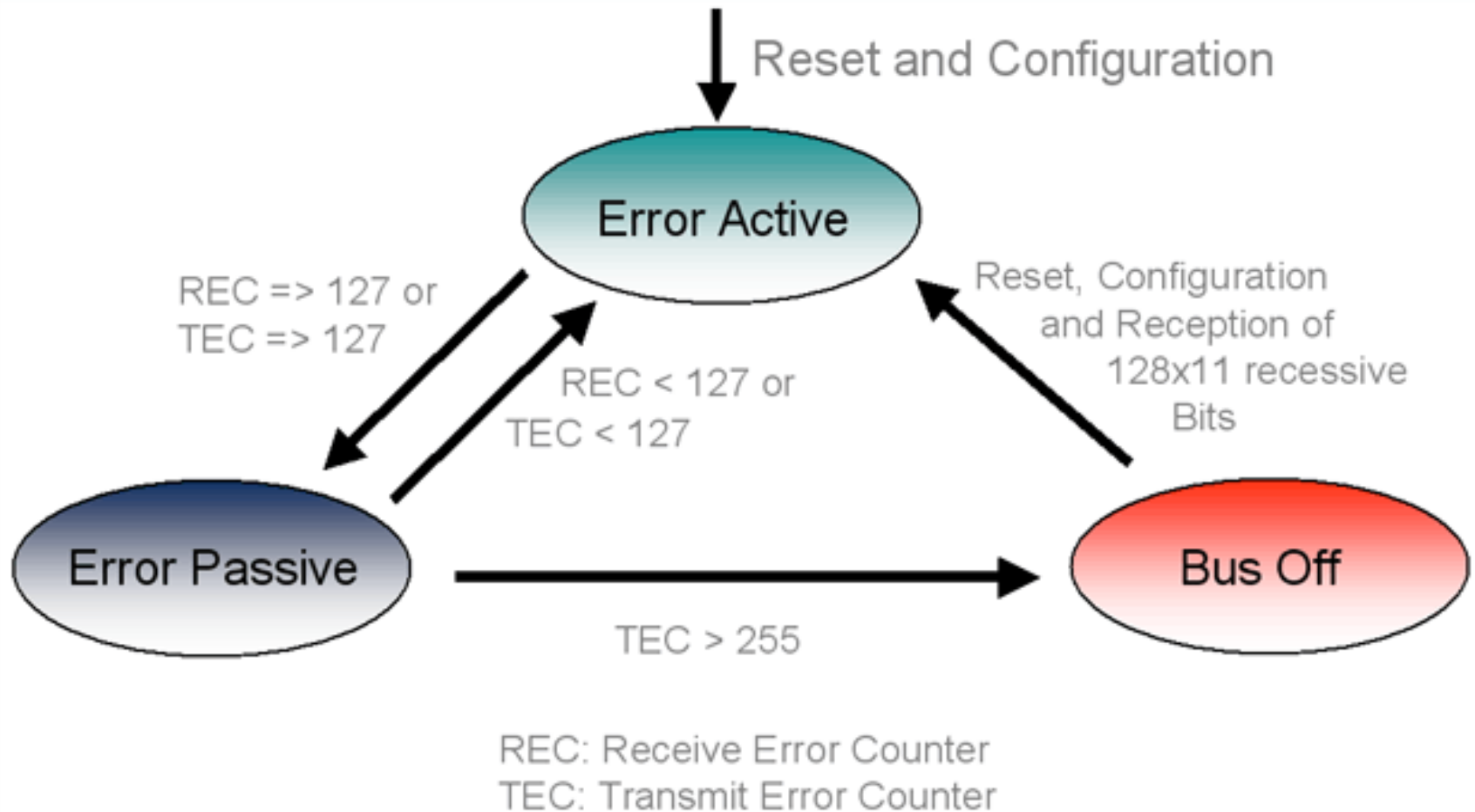
$$\rho < 4.7 \times 10^{-11} \times \text{error rate}$$

Example: 1 bit error each 0,7 s, 500 kbit/s,
working 8 h/day at 365 days/year.

Statistical average:

1 undetected error in 1000 years!

CAN節點的錯誤狀態



錯誤狀態的種類

1. 主動錯誤狀態：
節點處於正常狀態
2. 被動錯誤狀態：
可接收資料、接收時不能傳送錯誤通知。
3. 匯流排關閉狀態
禁止傳送/接收資料。

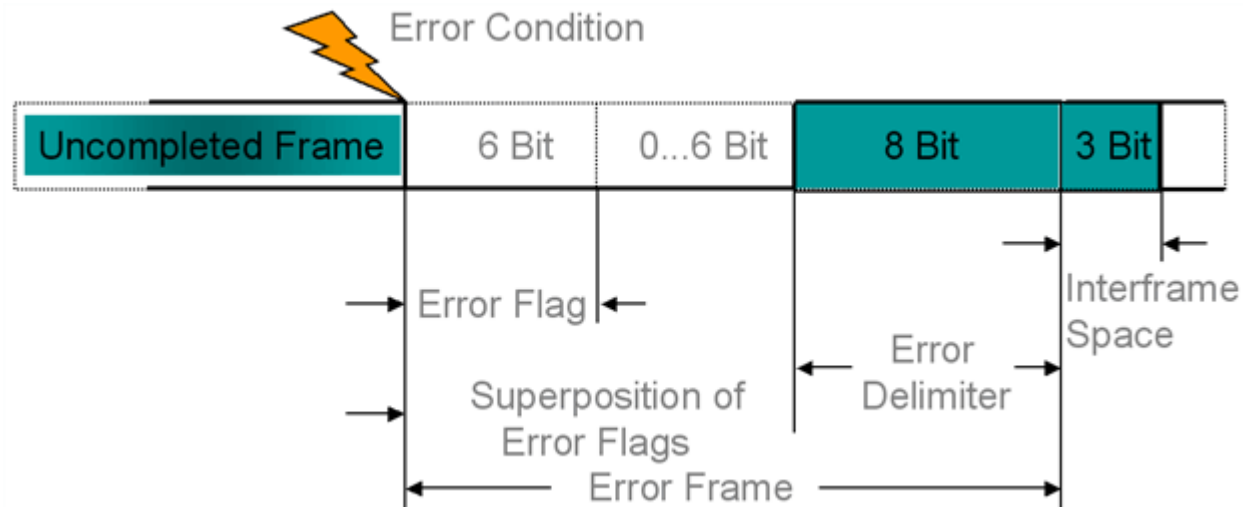
節點錯誤狀態	傳送錯誤計數值(TEC)	接收錯誤計數值(REC)
主動錯誤狀態	0-127	且 0-127
被動錯誤狀態	128-255	或 128-255
匯流排關閉狀態	256	----

匯流排關閉狀態

- 當發送錯誤計數器超過255的上限，匯流排狀態位元(bus status bit)被設定為邏輯1(匯流排關閉)，CAN控制器設定復位請求位元(reset request bit)為邏輯1，產生錯誤中斷直到CPU清除復位請求位元。
- CAN控制器將等待規定的最小時間(出現128個匯流排空閒信號)。之後，匯流排狀態位元被清除(匯流排上線)，錯誤狀態位元被設定為邏輯0(OK)，錯誤計數器復位並產生一個錯誤中斷。

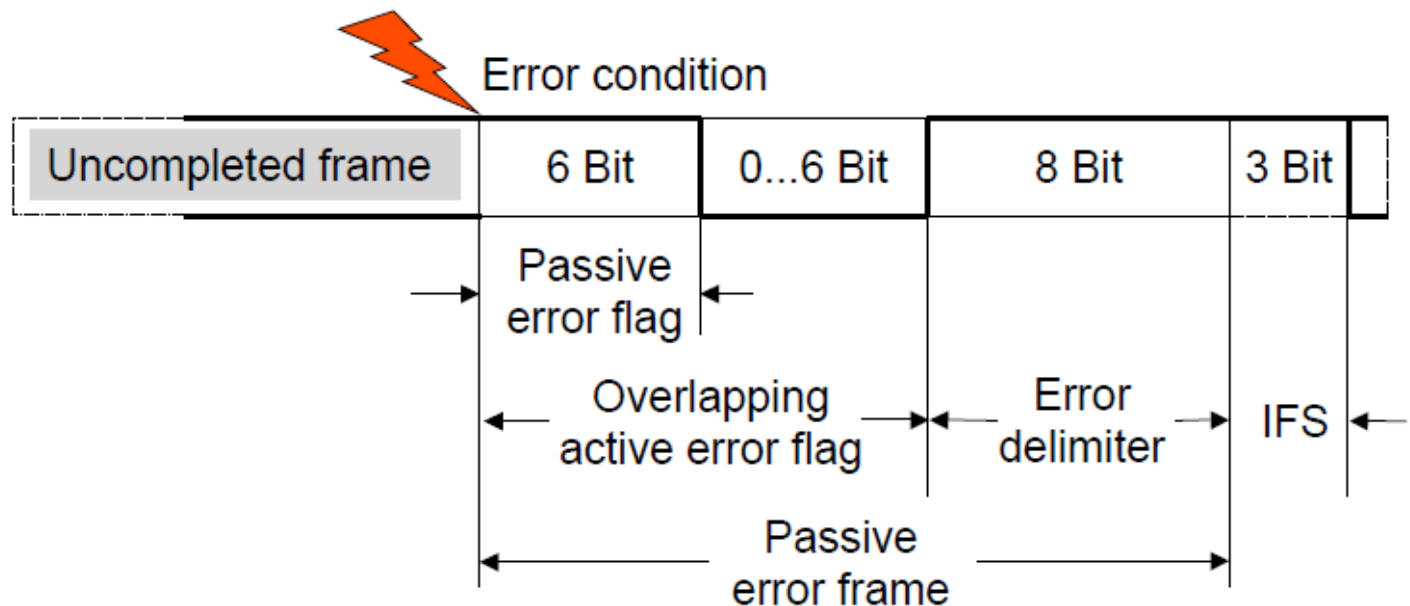
CAN bus錯誤處理-主動訊息格式

- 在發生bus error，產生6bits的error flag (dominant)，其他節點也會產生error flag，延遲0-6bits。
- 經過8bits error delimiter，3bits interframe 。
- 重傳



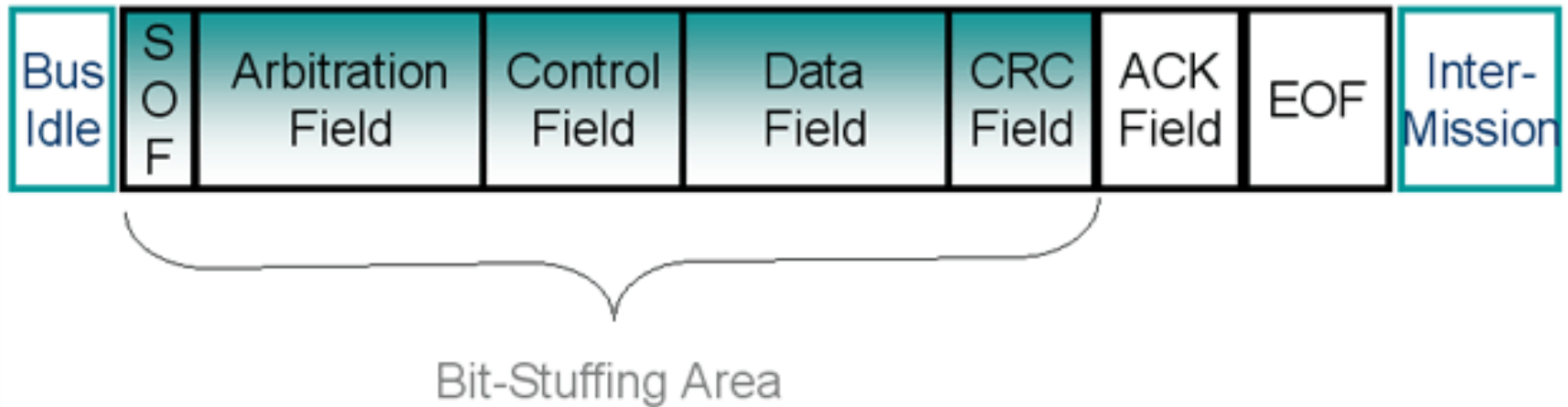
CAN bus 錯誤處理-被動訊息格式

- 當發生bus error，產生6bits的error flag (recessive)，其他節點也會產生error flag，延遲0-6bits。
- 經過8bits error delimiter，3bits interframe。
- 重傳



CAN bus錯誤處理-位元填塞錯誤

- 在SOF和CRC之間如果有連續相同的6個0或1，產生填塞錯誤。
- 在偵測到錯誤的下一個位元，送出error frame。



CAN bus 填塞規則

- 當連續出現相同的5bits時，傳送時加入1bit相反位元，接收時移除

bit-sequence to be transmitted



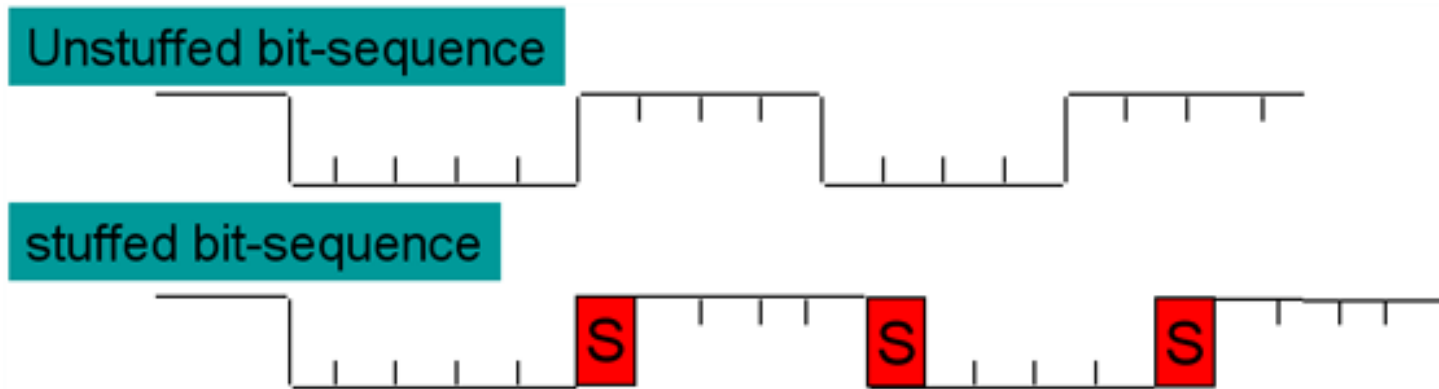
stuffed bit-sequence



de-stuffed bit-sequence received



CAN bus 使用者位元對填塞位元比

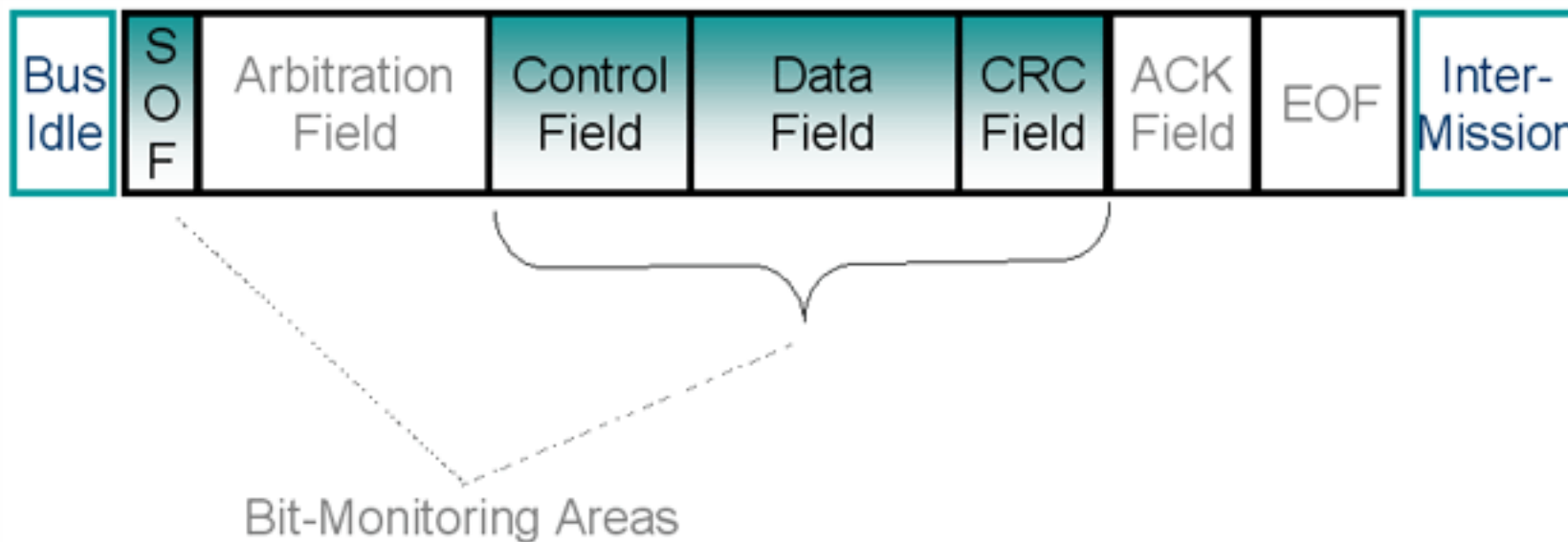


worst-case calculation for frames in standard format:

$$s_{\max} = (34 + 8 \text{ dlc} - 1) : 4 \approx 8 + 2 \text{ dlc}$$

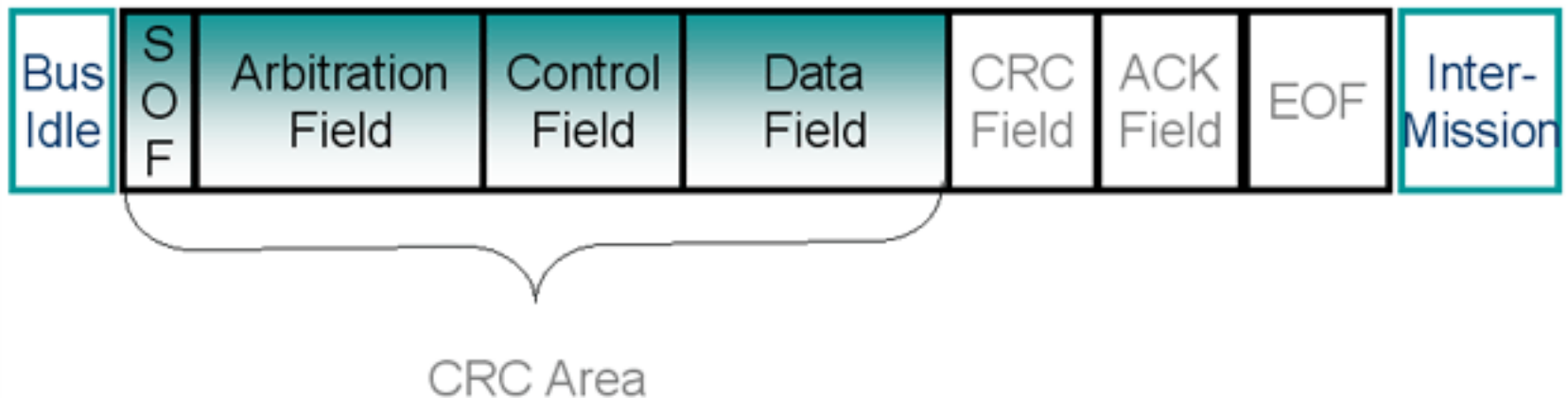
CAN bus錯誤處理- 位元錯誤

- 位元錯誤：匯流排傳送dominant但偵測到recessive；或傳送recessive但偵測到dominant。
- 偵測到錯誤的下一個位元，送出error frame。

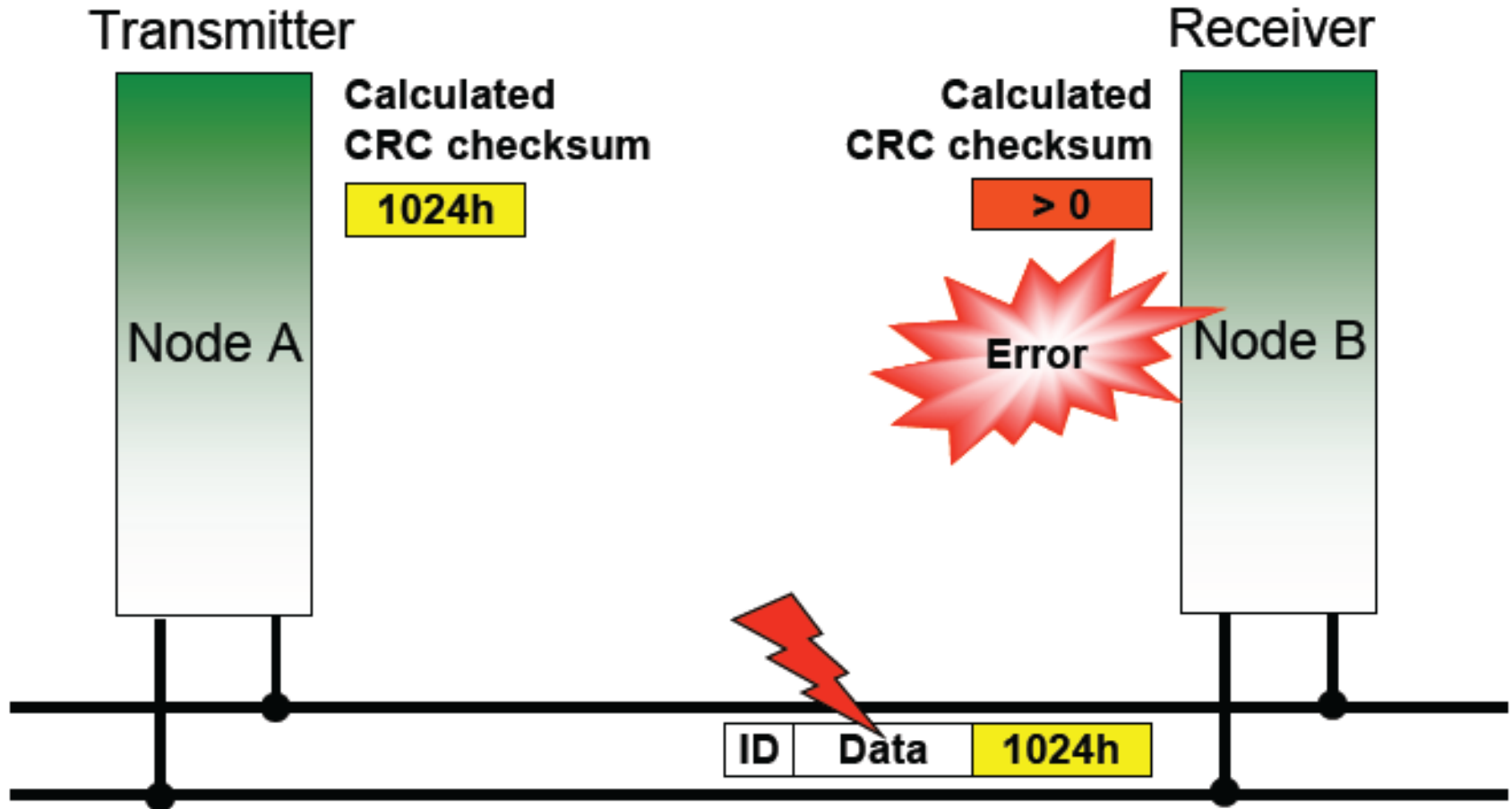


CAN bus錯誤處理– CRC錯誤

- 每個節點收到CRC Area資料後，計算CRC的值；與接收的CRC Field比較。
- 偵測到錯誤的CRC，在ACK Field之後，送出error frame。



CAN bus錯誤處理 – CRC錯誤



CAN bus錯誤處理- 確認錯誤

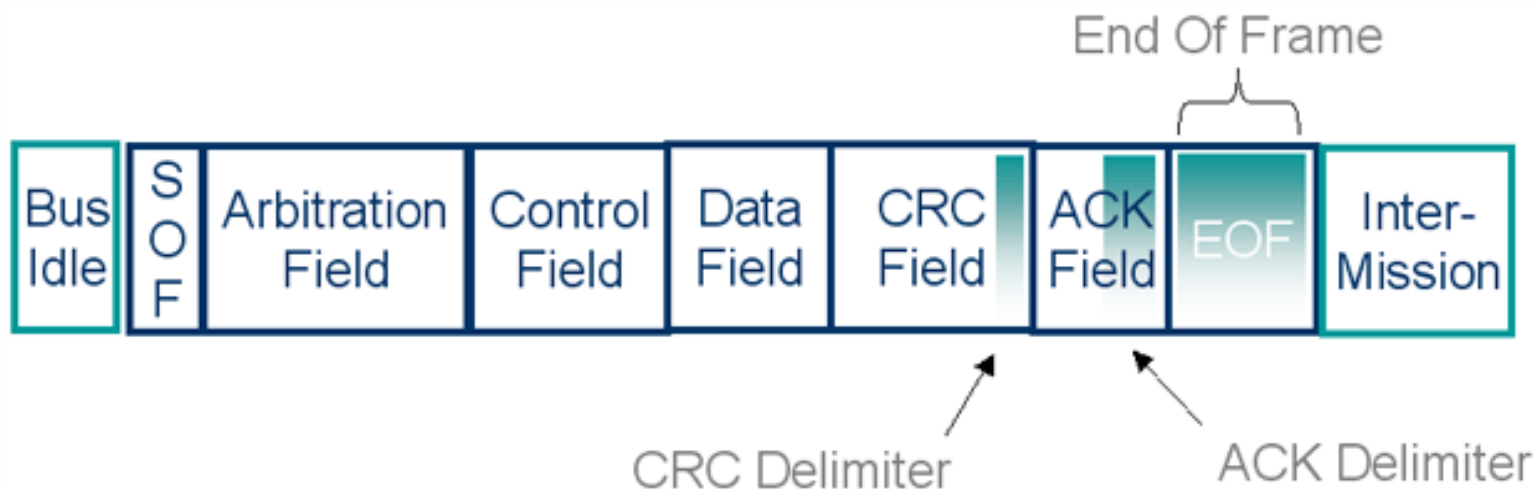
- 傳送者偵測到至少有一節點在ACK Field送出dominant(0)，表示正確接收。否則在ACK delimiter之後，送出error frame。



An ACK Error has to be detected by a Transmitter whenever it does not monitor a dominant bit during ACK Slot

CAN bus錯誤處理-格式錯誤

- 在CRC Delimiter、ACK Delimiter、EOF應該都是successive(1)
 - 如果偵測到**dominant(0)**，在下一個位元，送出error frame。
- 如果是EOF的最後一個位元為**dominant(0)**，接收端送出**overload frame**，傳送節點送出**error frame**。

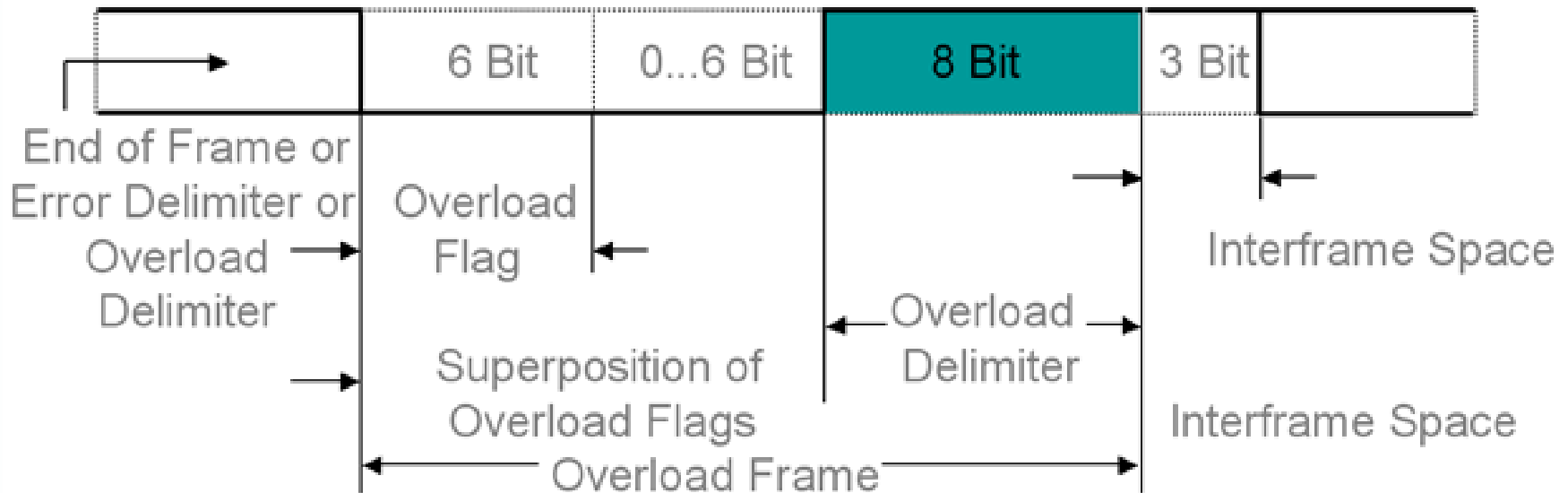


CAN bus錯誤處理-類別

錯誤種類	輸出時脈
Bit Error Bit-Stuffing Error Ack Error Form Error	從檢測出錯誤的下一個位元開始輸出錯誤旗標
CRC Error	在ACK delimiter之後的下一個位元開始輸出錯誤旗標

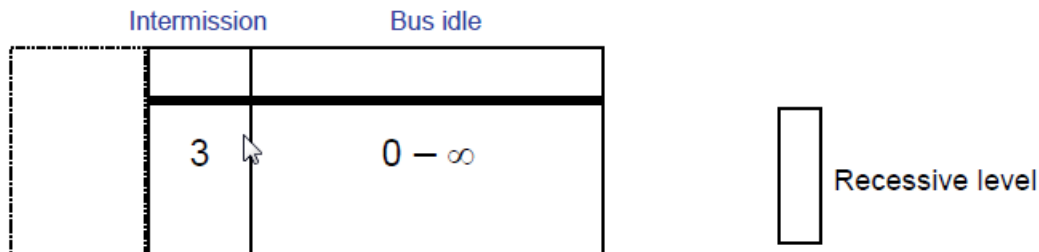
CAN bus錯誤處理- 過載通訊格式

- 當節點內部還不能夠接收下一個訊息，或
- 在Intermission的前二個位元為dominant.
- 過載通訊格式由節點產生

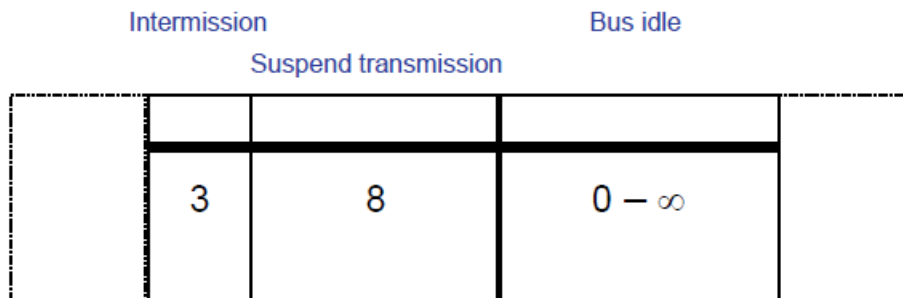


通訊格式間隔

- 用於分隔資料通訊格式，遠端通訊格式。
- 過載通訊格式和錯誤通訊格式之前不能加入通訊格式間隔 3bits recessive(1)。

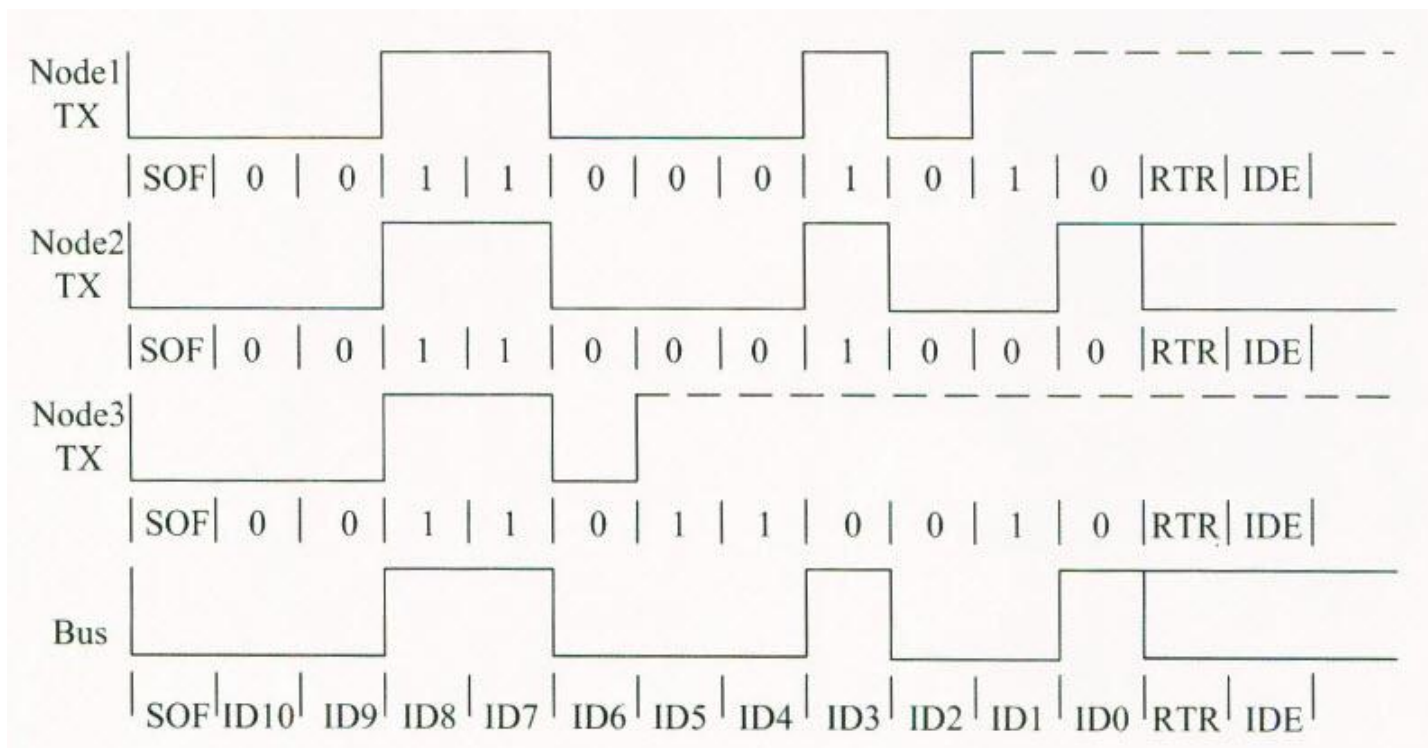


<When the immediately preceding message transmit unit is in an error-passive state>



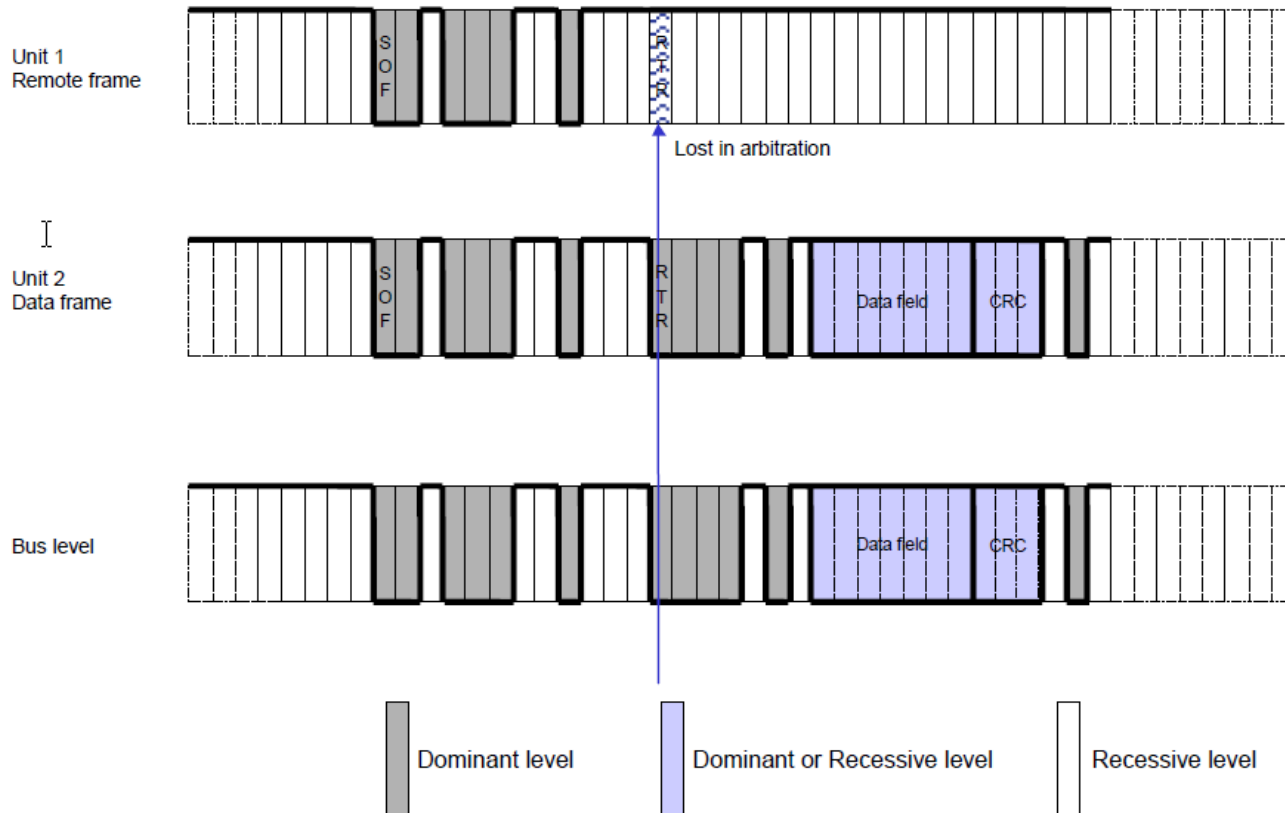
優先順序

- 若多個節點同時傳送，在仲裁欄位逐一比對優先權，位元為dominant(0)可優先傳送。



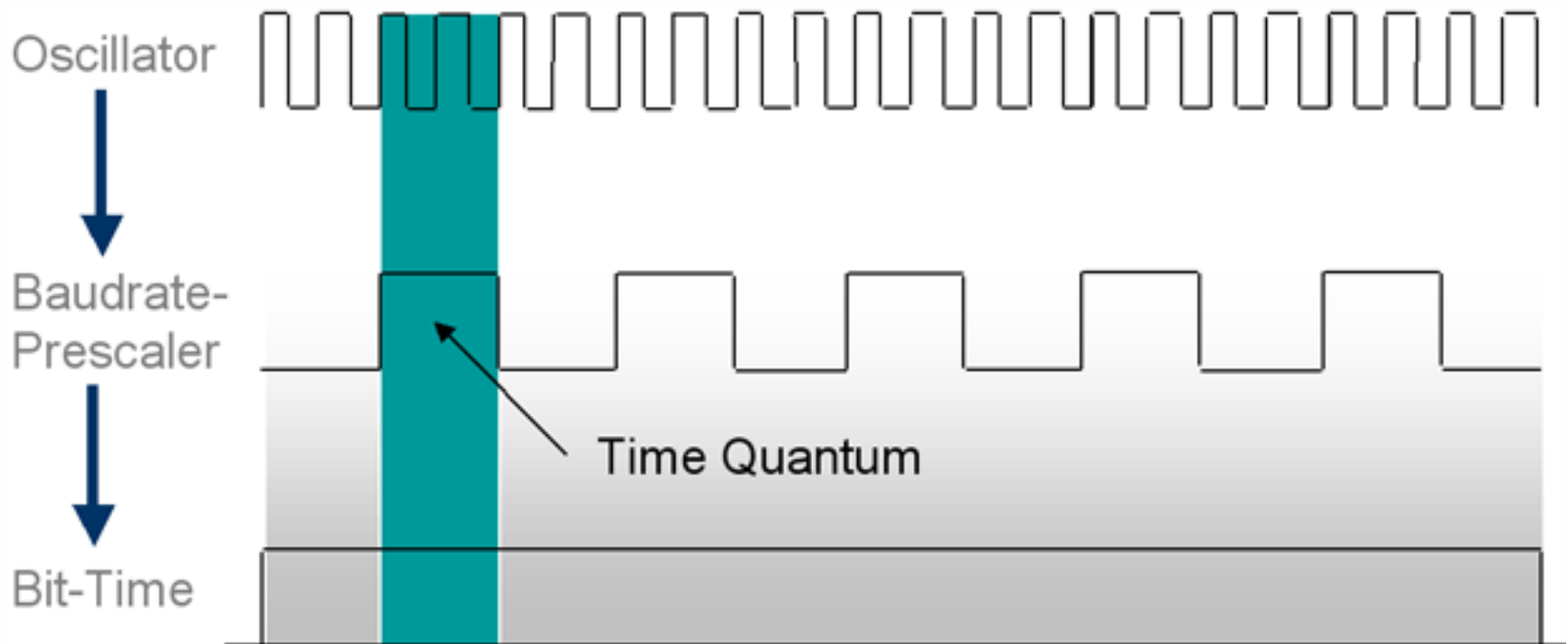
資料通訊和遠端通訊同時傳送

- 由於二者的ID相同，優先權相同，比較到RTR位元時，資料通訊先傳送。



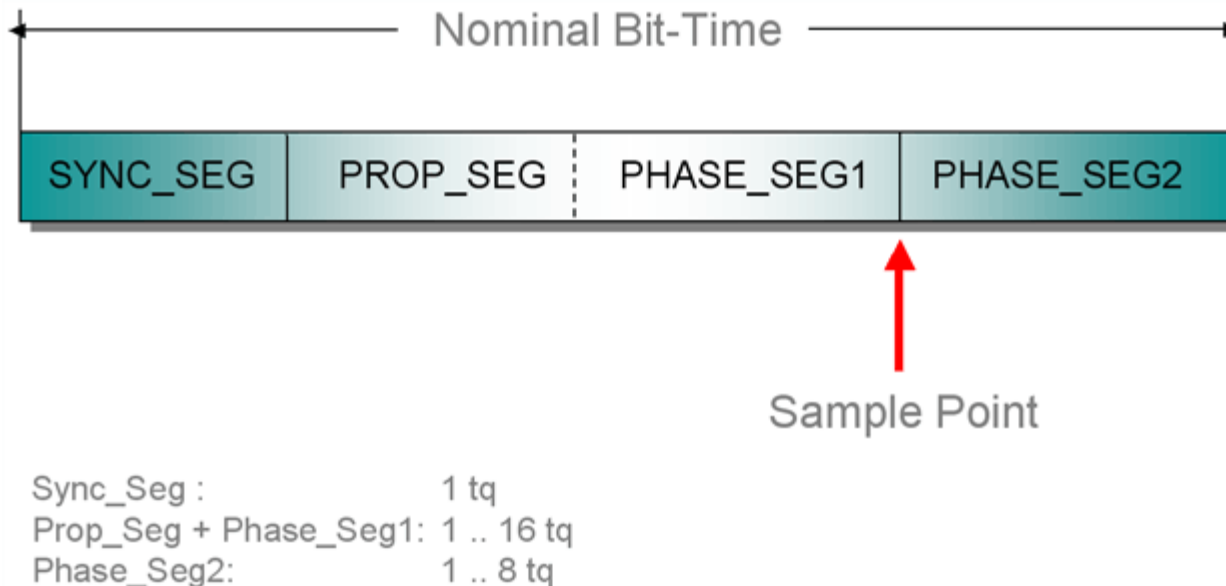
CAN bus 位元時脈

- 每一個位元最少有8，最多有25個Time Quanta.

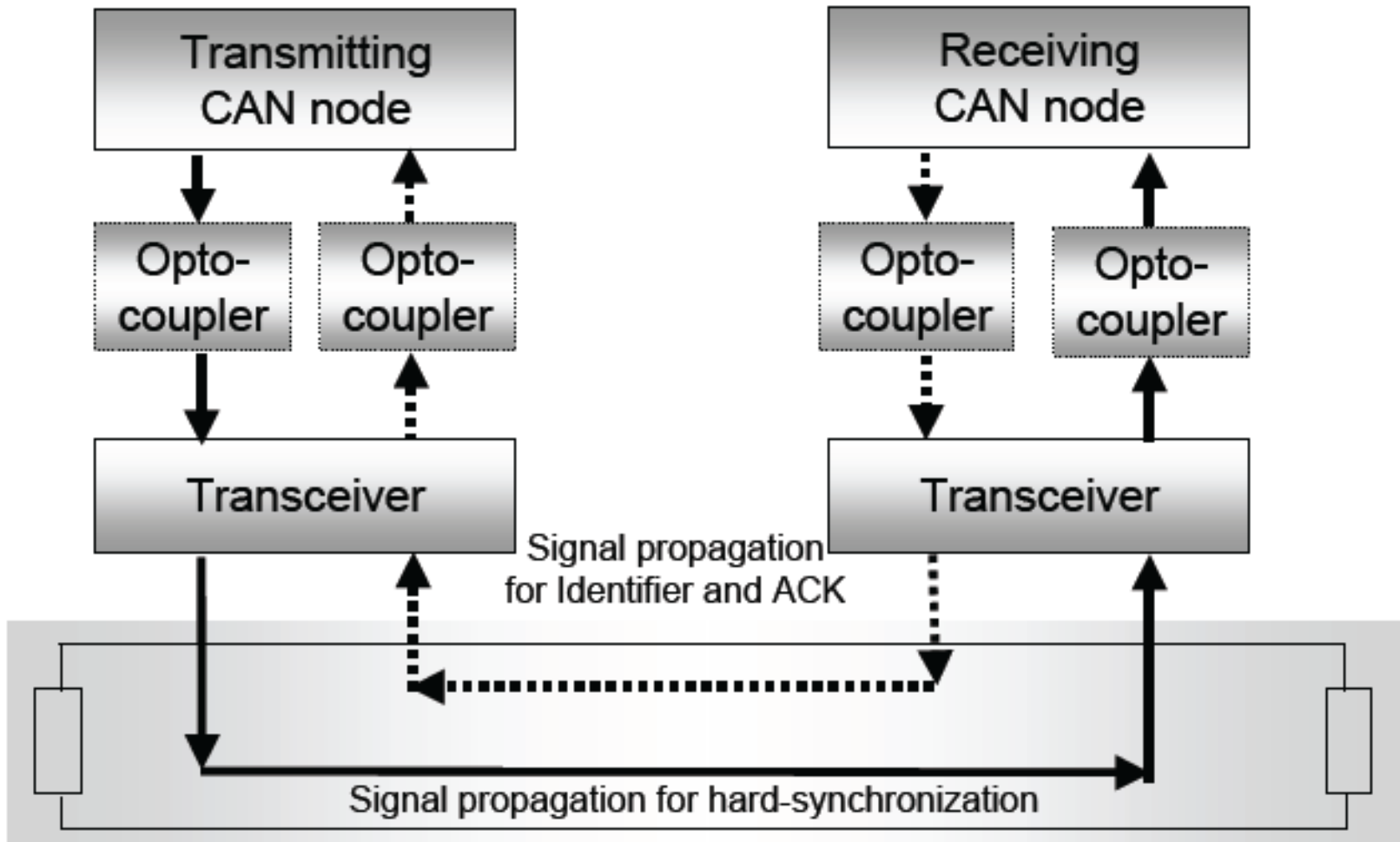


CAN bus 次位元區段

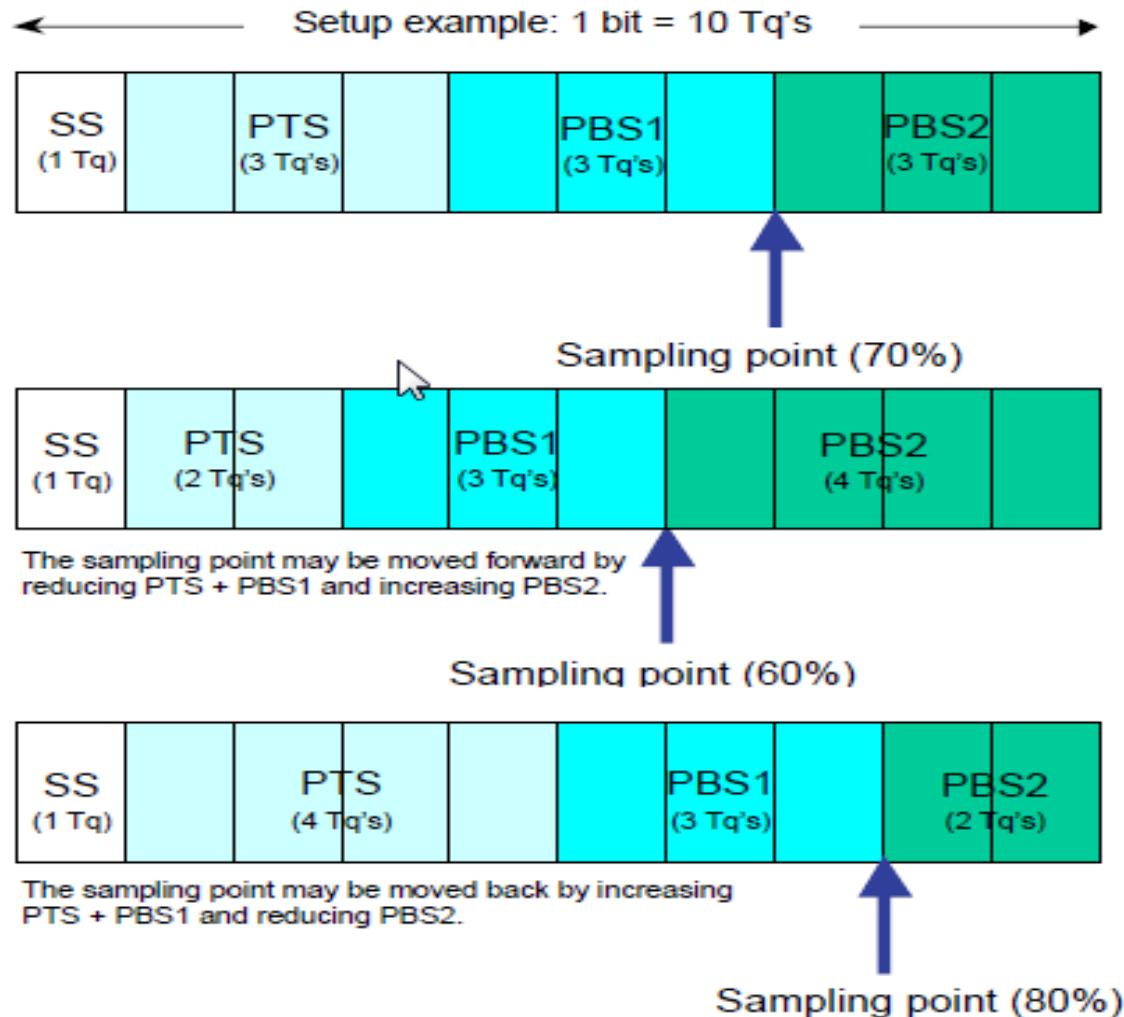
- SYNC_SEG 是1個 Time Quantum.
- PROP_SEG 可規劃為1,2, ... 8 TQ.
- PHASE_SEG1可規劃為1,2, ... 8 TQ
- PHASE_SEG2設定為 $(1-4TQ) \leq \text{PHASE_SEG1}$.



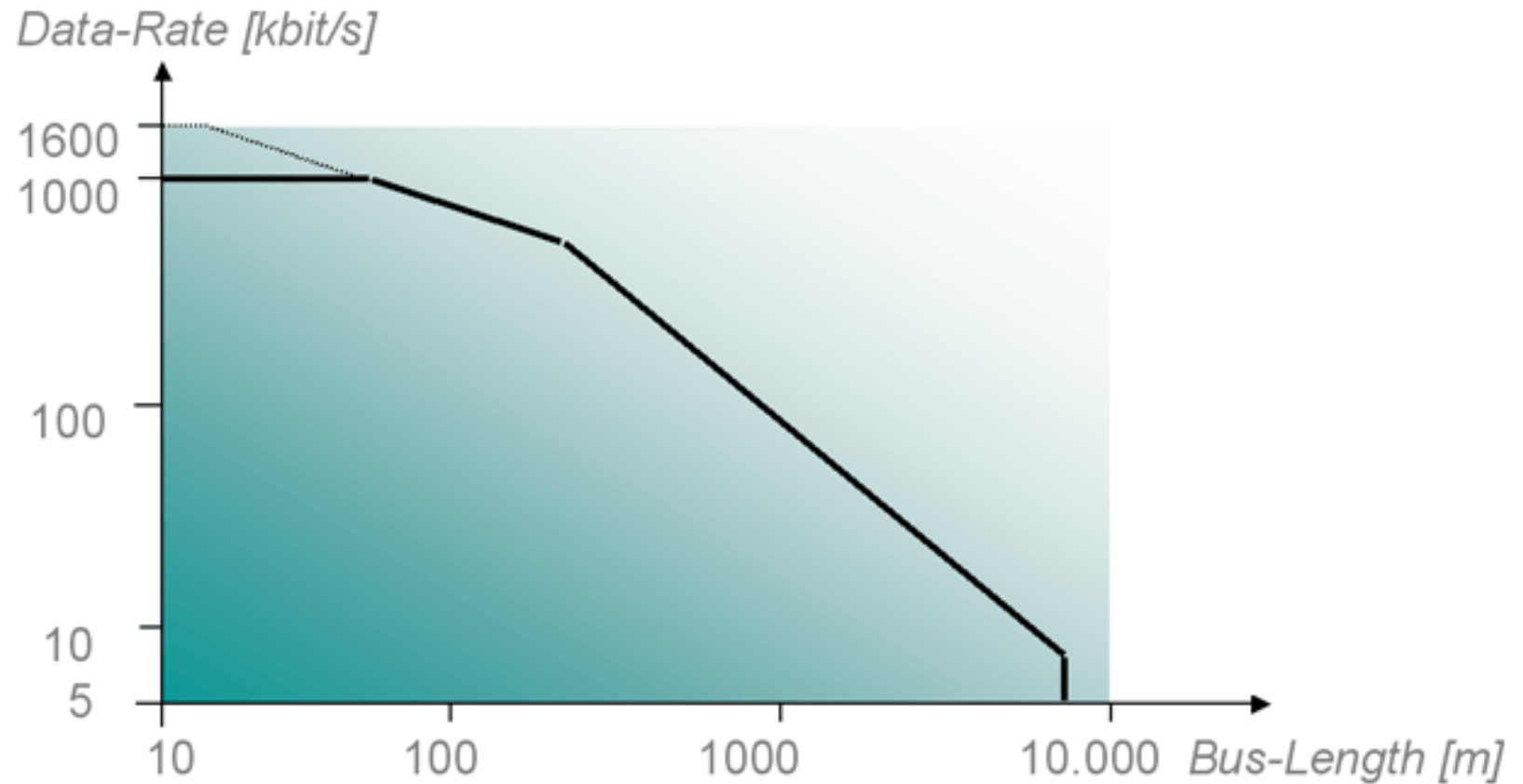
Signal propagation



CAN bus 次位元區段



CAN bus 資料率 / Bus-長度比



CAN 實際Bus長度

Bit Rate	Bus Length	Nominal Bit-Time
1 Mbit/s	30 m	1 μ s
800 kbit/s	50 m	1,25 μ s
500 kbit/s	100 m	2 μ s
250 kbit/s	250 m	4 μ s
125 kbit/s	500 m	8 μ s
62,5 kbit/s	1000 m	20 μ s
20 kbit/s	2500 m	50 μ s
10 kbit/s	5000 m	100 μ s

CAN bus ISO 高速收發器

Manufacturer	Bosch	Mietec	Philips Semiconductors	Philips Semiconductors	SGS-Thomson	Temic (Siliconix)	Unitrode
type no.	CF150B	MTC-3054	82C250	82C251	L9615	SI9200EY	UC5350
data rate max. [Mbd]	0.5	1	1	1	0.5	1	1
short circuit [V]	-5...+36	-3...+65	-8...+18	-36...+36	-5...+36	GND...+16	-8...+36
transient [V]	-200...+200	-200...+200	-150...+100	-200...+200	-200...+200	-60...+60	-150...+100
ESD [kV]	2	2	2	2.5	2	2	2
thermal shutdown	(1,2)	n.a.	yes	yes	(1, 2)	yes	yes
slope control	on/off	variable	variable	variable	on/off	none	variable
CMR [V]	-2...+7 (3)	-7...+12	-7...+12	-7...+12	-2...+7 (3)	-2...+7	-25...+18
delay [ns]	230	100	170	170	230	120 (4)	100 (4)
fan out (5)	32	32	64 (110)	110	32	32	n.a. (6)
supply current [mA]	<80	110	<70	<80	<80	70	70
stand-by current [μ A]	n.a.	300	<170	<250	n.a.	n.a.	1000
packaging	SOIC-8	SOP-16	SO-8, DIP-8	SO-8, DIP-8	SO-8	SO-8	SOIC-8,DIL-8

(1) short circuit detection and shutdown (2) overvoltage detection and shutdown (3) measured CMR: -12...+18V

(4) 50 ns for transmit (5) higher fan out possible, depends on application

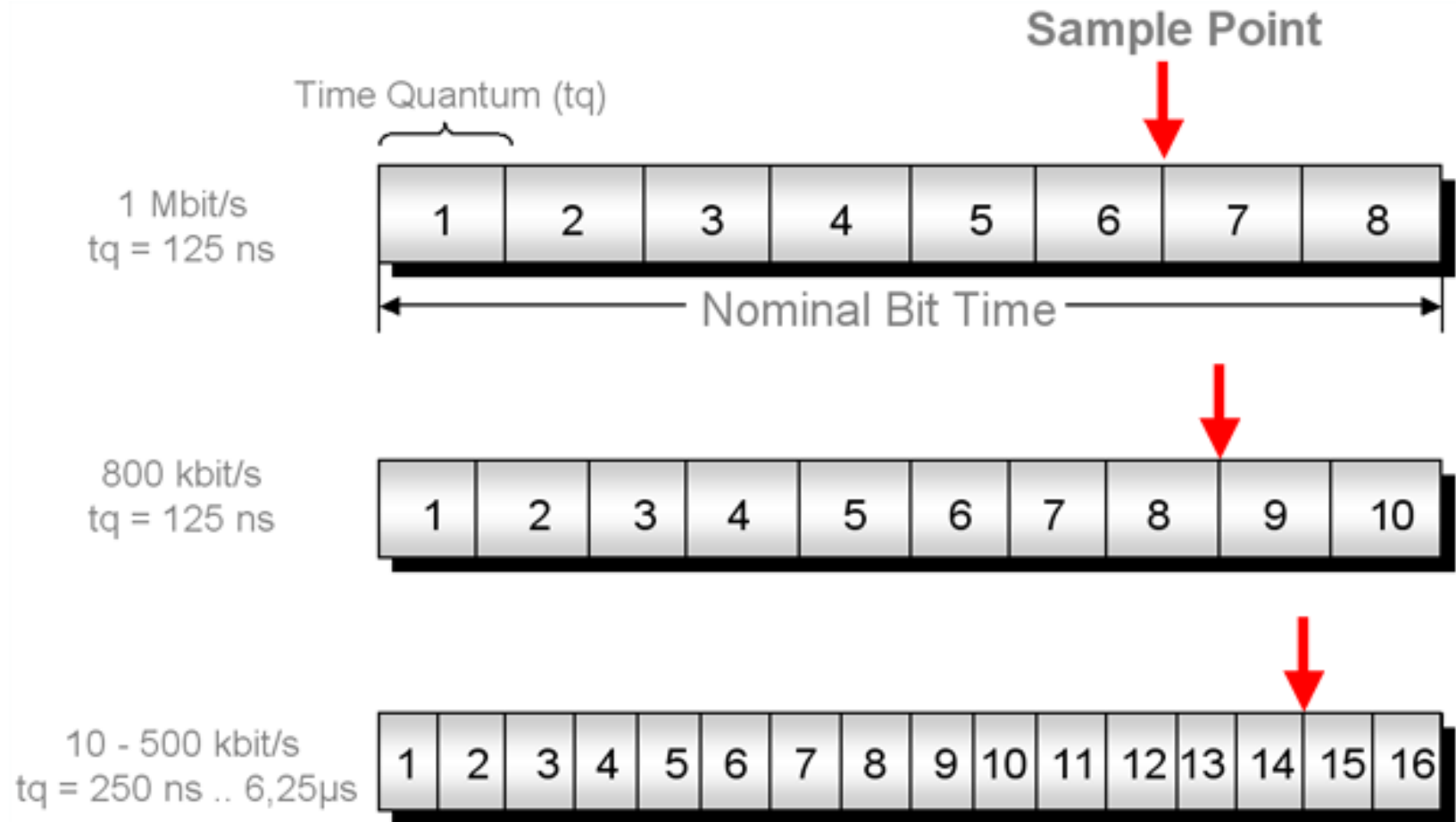
(6) under specific circumstances more than 128 n.a.: data not available

Remarks: 82C250 and UC5350 are pin compatible; L9615 is licenced by Bosch

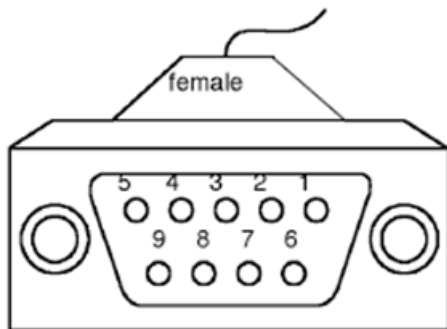
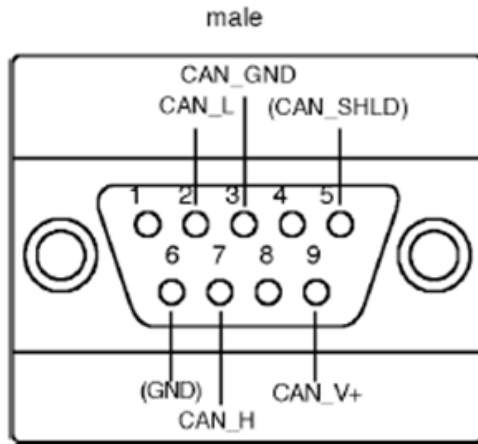
CAN bus CiA DS-102 Baudrates

Bit rate Bus length ⁽¹⁾	Nominal bit time t_b	Number of time quanta per bit	Length of time quantum t_q	Location of sample point	BTR 0 at 16 MHz (82C200)	BTR 1 at 16 MHz (82C200)
1 Mbit/s 25 m	1 μ s	8	125 ns	6 t_q (750ns)	00h	14h
800 kbit/s 50 m	1.25 μ s	10	125 ns	8 t_q (1 μ s)	00h	16h
500 kbit/s 100 m	2 μ s	16	125 ns	14 t_q (1.75 μ s)	00h	1Ch
250 kbit/s 250 m ⁽²⁾	4 μ s	16	250 ns	14 t_q (3.5 μ s)	01h	1Ch
125 kbit/s 500 m ⁽²⁾	8 μ s	16	500 ns	14 t_q (7 μ s)	03h	1Ch
50 kbit/s 1000 m ⁽³⁾	20 μ s	16	1.25 μ s	14 t_q (17.5 μ s)	09h	1Ch
20 kbit/s 2500 m ⁽³⁾	50 μ s	16	3.125 μ s	14 t_q (43.75 μ s)	18h	1Ch
10 kbit/s 5000 m ⁽³⁾	100 μ s	16	6.25 μ s	14 t_q (87.5 μ s)	31h	1Ch

CAN bus CiA DS-102 Bit-Timing



CAN bus CiA DS-102 Pin Assignment



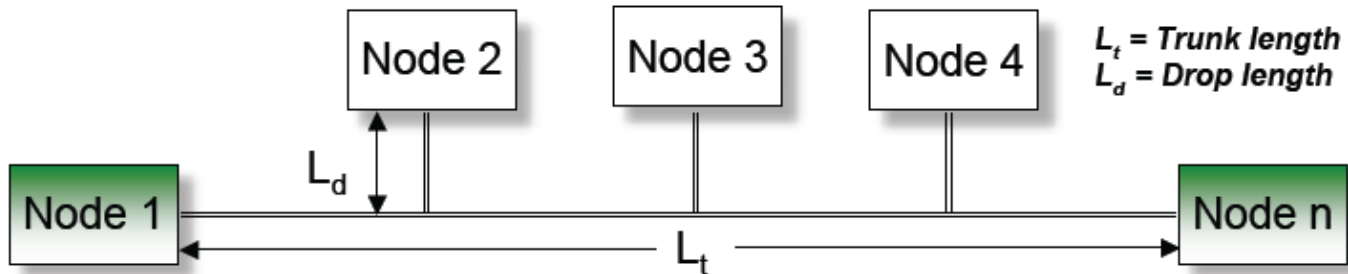
9-pin D-Sub: DIN 41652

Pin	Signal	Description
1	-	Reserved
2	CAN_L	CAN_L bus line dominant low
3	CAN_GND	CAN ground
4	-	Reserved
5	(CAN_SHLD)	Optional CAN shield
6	GND	Optional ground
7	CAN_H	CAN_H bus line dominant high
8	-	Reserved
9	(CAN_V+)	Optional CAN external supply

CAN bus DC 特性

Bus Length	Bus Cable		Termination Resistance	Max. Baudrate
	Length-Related Resistance	Bus-Line Cross-Section		
0 .. 40 m	70 mΩ/m	0.25 mm ² .. 0.34 mm ² AWG23, AWG22	124 Ω (1%)	1 Mbit/s at 40 m
40 .. 300 m	<60 mΩ/m	0.34 mm ² .. 0.6 mm ² AWG22, AWG20	127 Ω (1%)	500 Kbit/s at 100 m
300 .. 600 m	<40 mΩ/m	0.5 mm ² .. 0.6 mm ² AWG20	150 Ω to 300 Ω	100 Kbit/s at 500 m
600 m .. 1 km	<26 mΩ/m	0.75 mm ² .. 0.8 mm ² AWG 18	150 Ω to 300 Ω	50 Kbit/s at 1k m

CAN bus ISO 11898-2 拓樸



Rules of thumb for the maximum length of a not terminated cable drop L_d and for the cumulative drop length L_{di} :

$$L_d < t_{\text{PROPSEG}} / (50 * t_p)$$

$$\sum_{i=1}^n L_{di} < t_{\text{PROPSEG}} / (10 * t_p)$$

t_{PROPSEG} : length of the propagation segment of the bit period
 t_p : specific line delay per length unit

Example: bit rate = 500 kbit/s; $t_{\text{PROPSEG}} = 11 * 125 \text{ ns} = 1375 \text{ ns}$; $t_p = 5 \text{ ns/m}$

$$L_d < 1375 \text{ ns} / (50 * 5 \text{ ns/m}) = 5,5 \text{ m}; \sum_{i=1}^n L_{di} < 1375 \text{ ns} / (10 * 5 \text{ ns/m}) = 27,5 \text{ m}$$

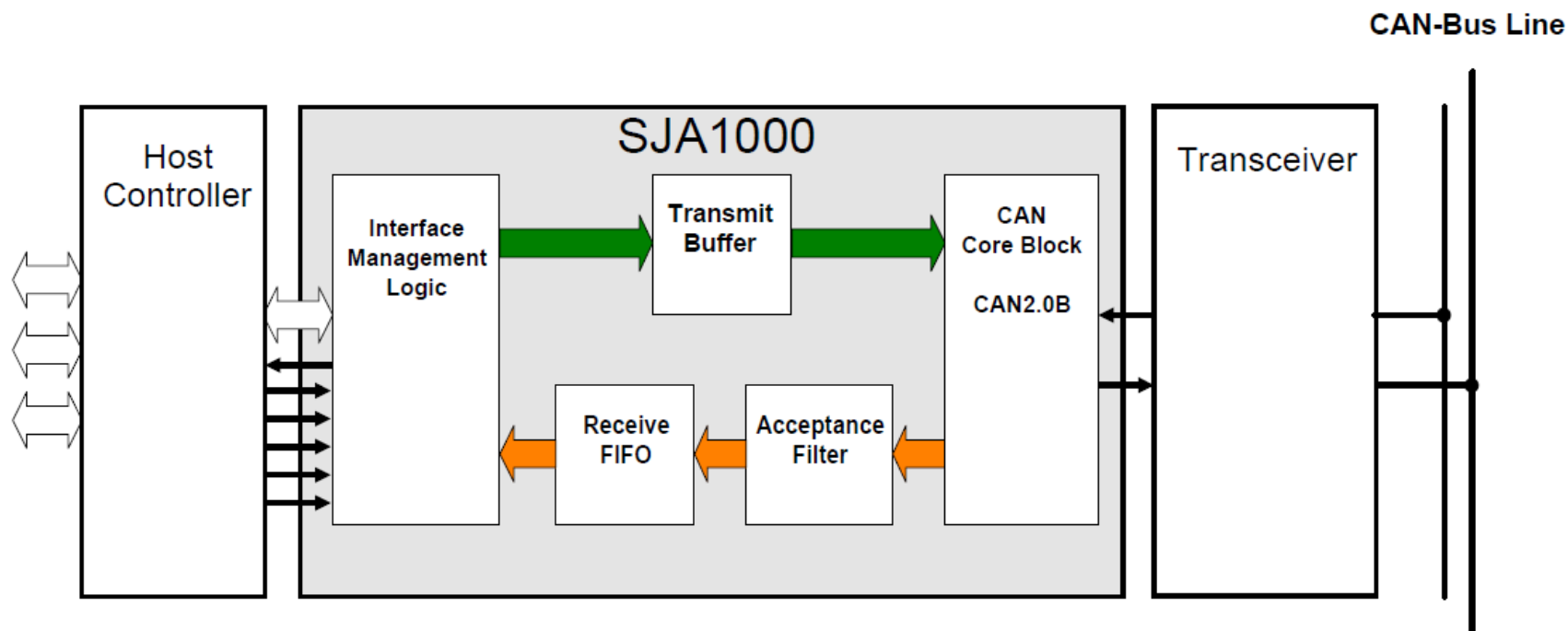
- 在1 Mbit/s連接纜線的長度 L_d 不能超過0.3 m.

Stub length limits

Data rate bus length	Sample Point (TQ)	Max. stub length	Accumulated Stub length
1 Mbit/s - 25 m	87,5% (125 ns)	1.5m	7.5m
800 kbit/s - 50 m	87,5% (125 ns)	2.5m	12.5m
500 kbit/s - 100 m	87,5% (125 ns)	5.5m	27.5m
250 kbit/s - 250 m	87,5% (250 ns)	11m	55m
125 kbit/s - 500 m	87,5% (500 ns)	22m	110m
50 kbit/s - 1000 m	87,5% (1,25 μ s)	55m	275m
20 kbit/s - 2500 m	87,5% (3,125 μ s)	137.5m	687.5m
10 kbit/s - 5000 m	87,5% (6,25 μ s)	275m	1375m

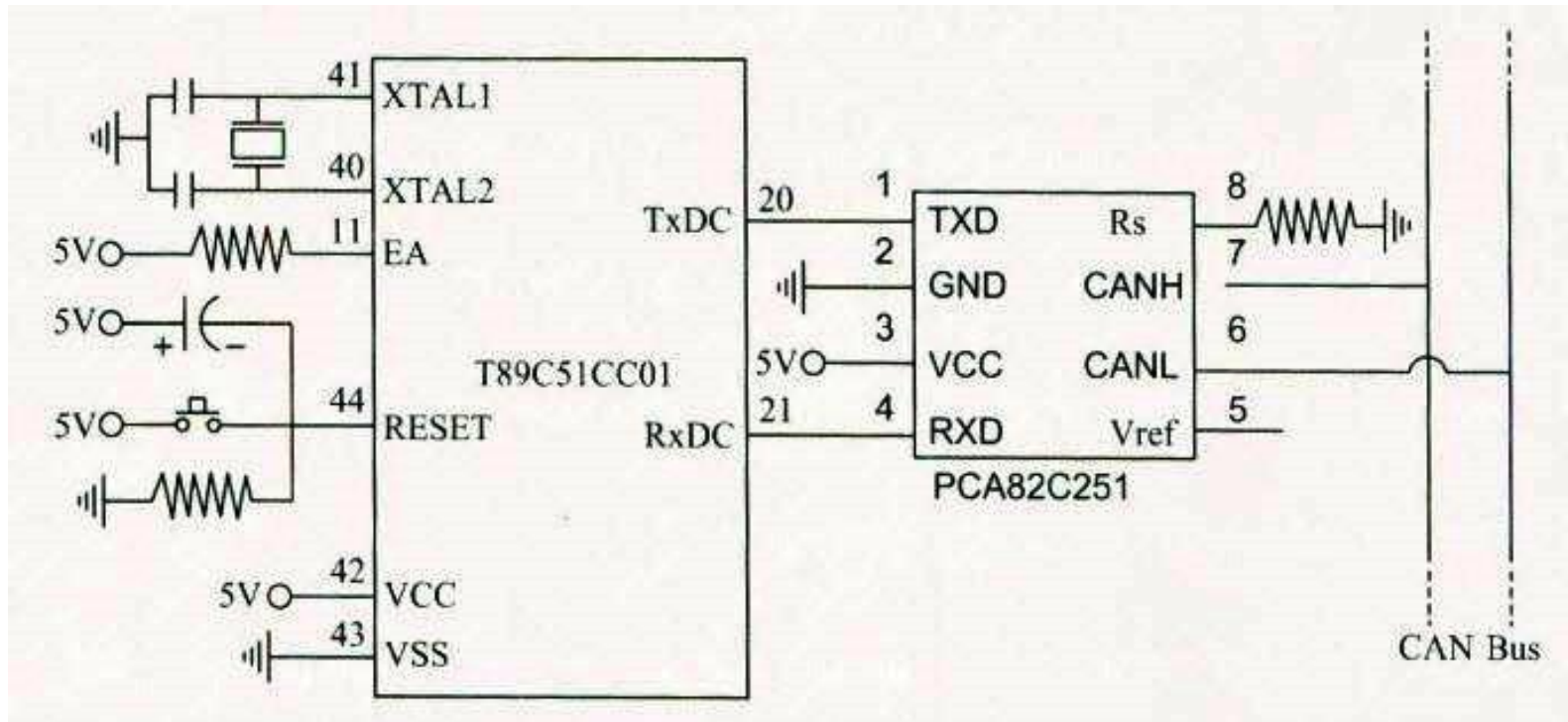
不含微處理器的CAN控制器

- 以CPU、CAN控制器(SJA1000)、CAN收發器(PC82C251)建構CAN ECU節點。



含CAN控制器的微處理器

- 使用含CAN控制器的微處理器(Atmel T89C51CC01: 8051+CAN)、CAN收發器(PCA82C251)建構CAN ECU節點。

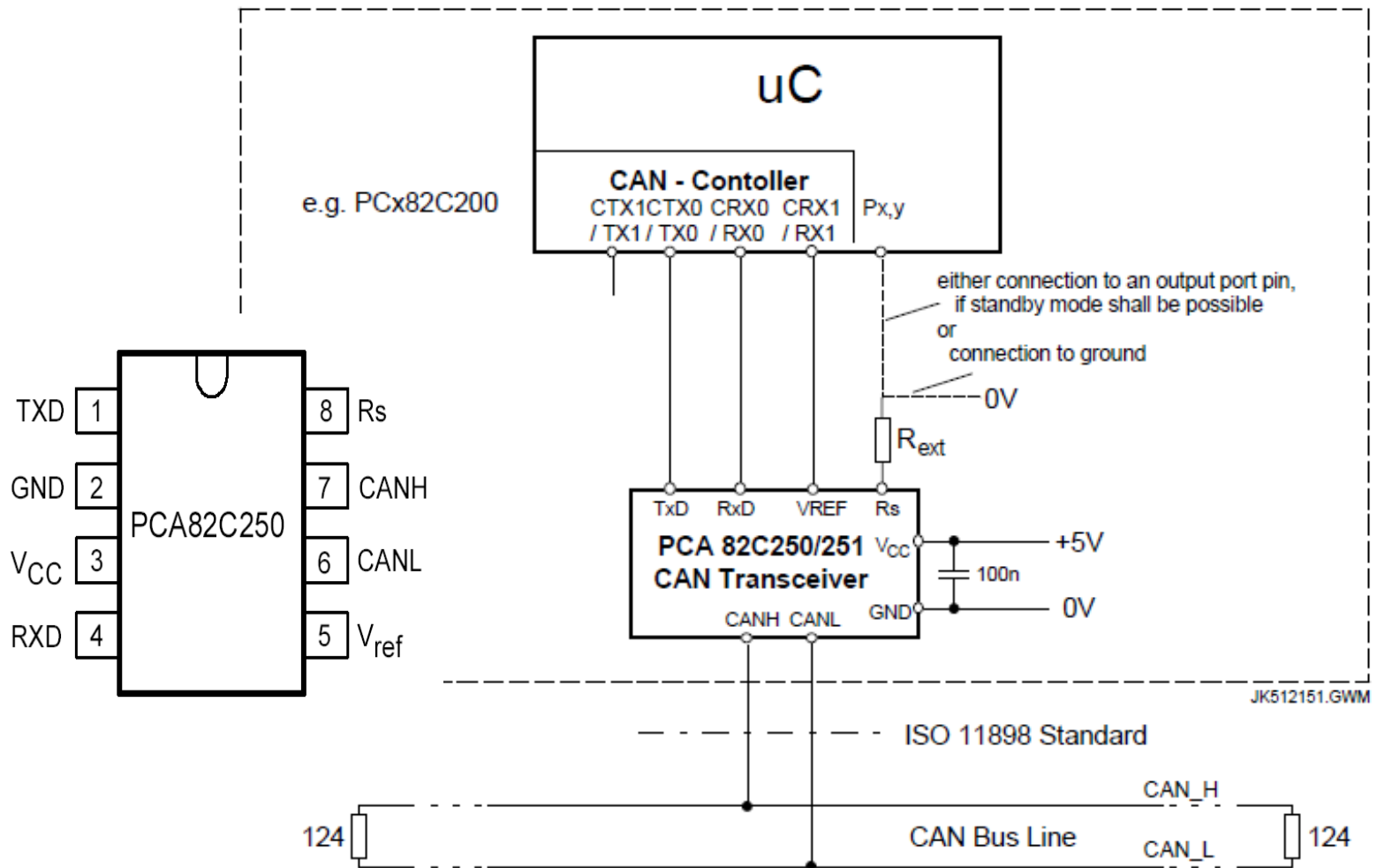


CAN匯流排收發器

PCA82C250/PCA82C251

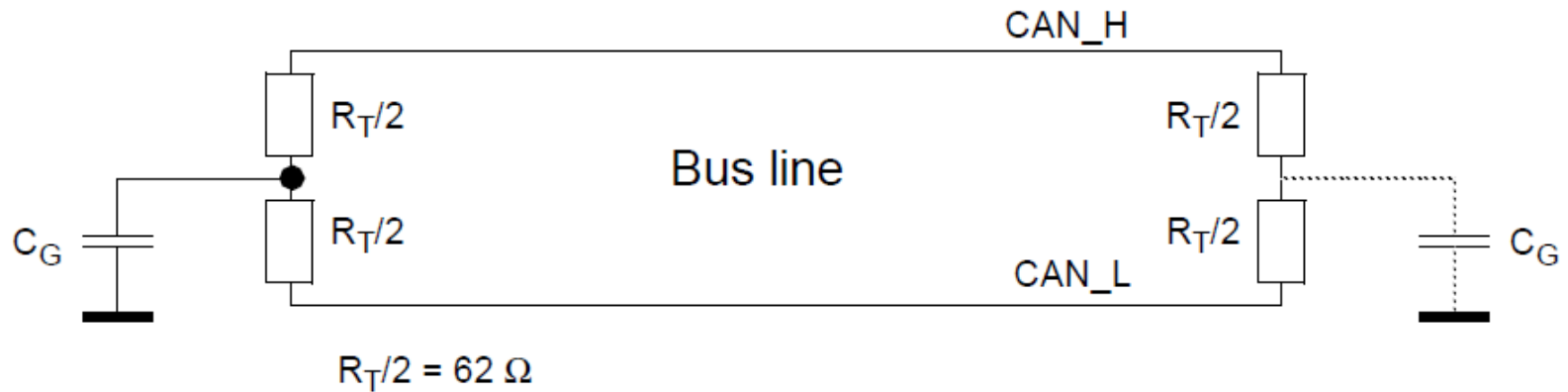
- PCA82C250/251是CAN協定控制器和實體匯流排的介面。此器件對匯流排提供差動發送能力，對CAN控制器提供差動接收能力。又稱為匯流排驅動器。
- PCA82C251有更高的擊穿電壓，而且在隱性狀態下拉電流更小，在掉電情況小匯流排的輸出特性有一定改善。
- 完全符合ISO11898標準。
- 高速率（最高達1Mbps）。
- 未上電的節點對匯流排無影響。
- 可連接110個節點。

PCA 82C250/251的應用



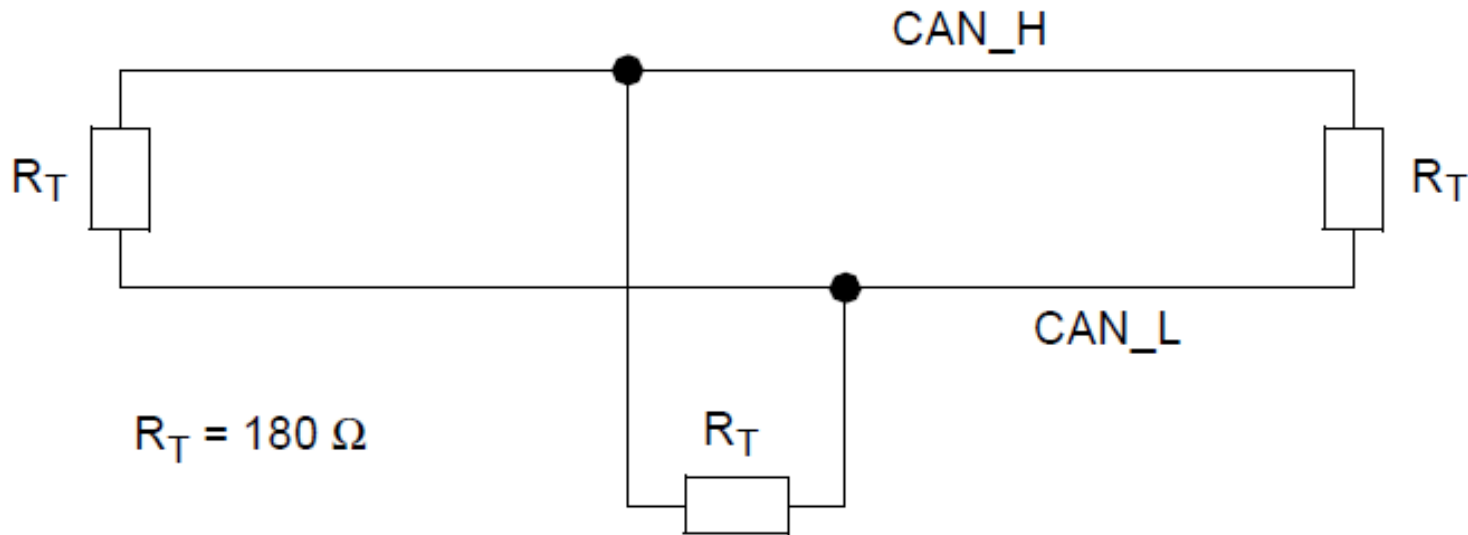
終端電阻的連接方式

- 如圖的接法將一個 124Ω 電阻分成二個 62Ω 電阻串聯，再連接一個 10nF - 100nF 的電容，可以增加EMC的特性。



終端電阻的連接方式

- 如果CAN Bus的連接不似線形，例如有3個分支的星狀，可以將其視為3個電阻的並聯，每個電阻值取 $180\ \Omega$ 。



獨立的CAN控制晶片 SJA1000

- SJA1000是一款獨立的CAN控制器，用於汽車和一般工業環境中的控制器局域網絡(CAN)。
- 和PCA82C200獨立CAN控制器電氣兼容；
- 擴展的接收緩衝器(64字節、先進先出FIFO)；
- 和CAN2.0B協議兼容；
- 同時支持11位和29位識別碼；
- 位速率可達1Mbits/s；
- PeliCAN模式擴展功能；
- 增強的環境溫度範圍(-40-+125°C)。

操作模式

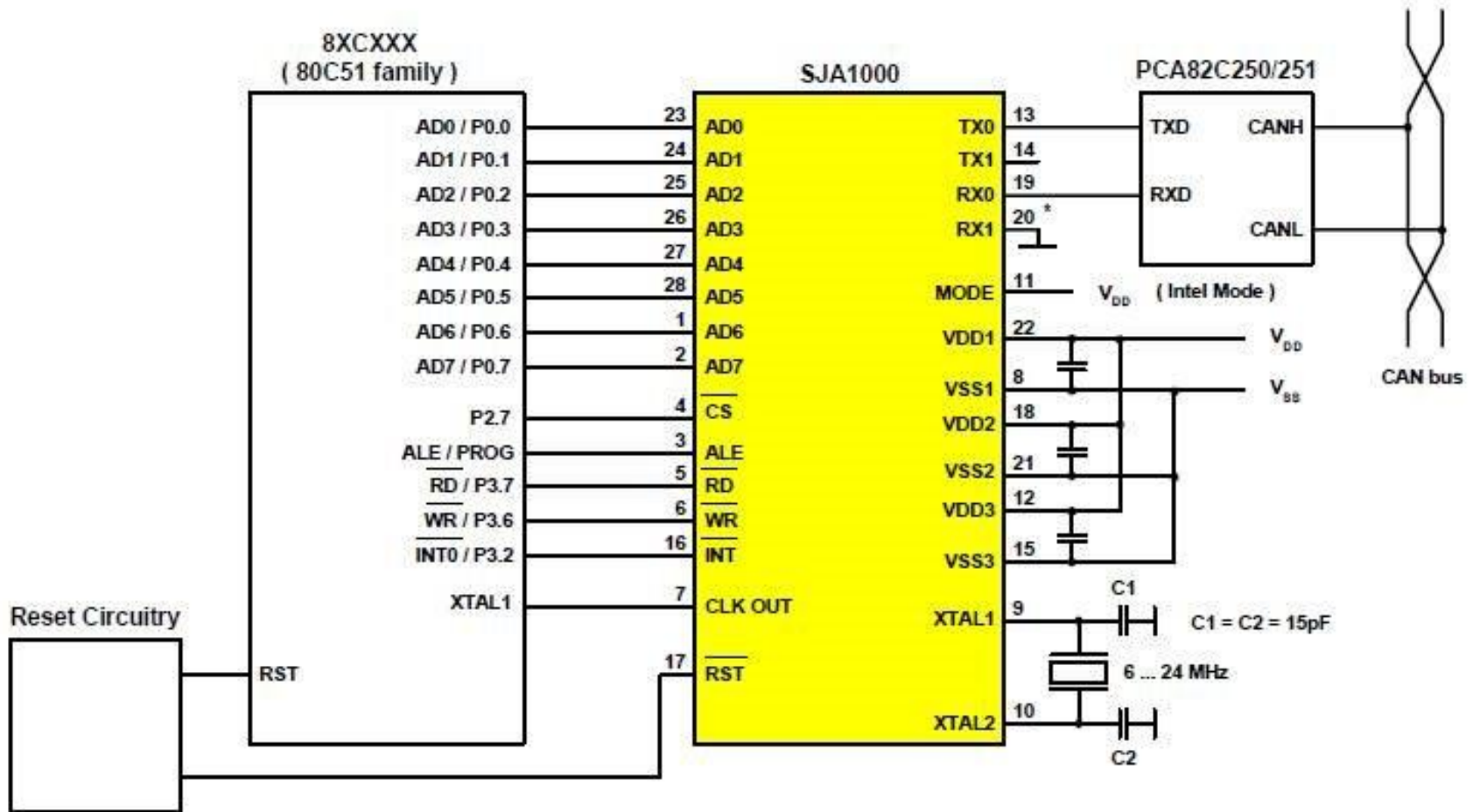
- BasicCAN模式

和PCA82C200相容。BasicCAN模式是上電後預設的操作模式，因此用PCA82C200開發的已有硬體和軟體，可以直接在SJA1000上使用而不用作任何修改。

- Pelican模式

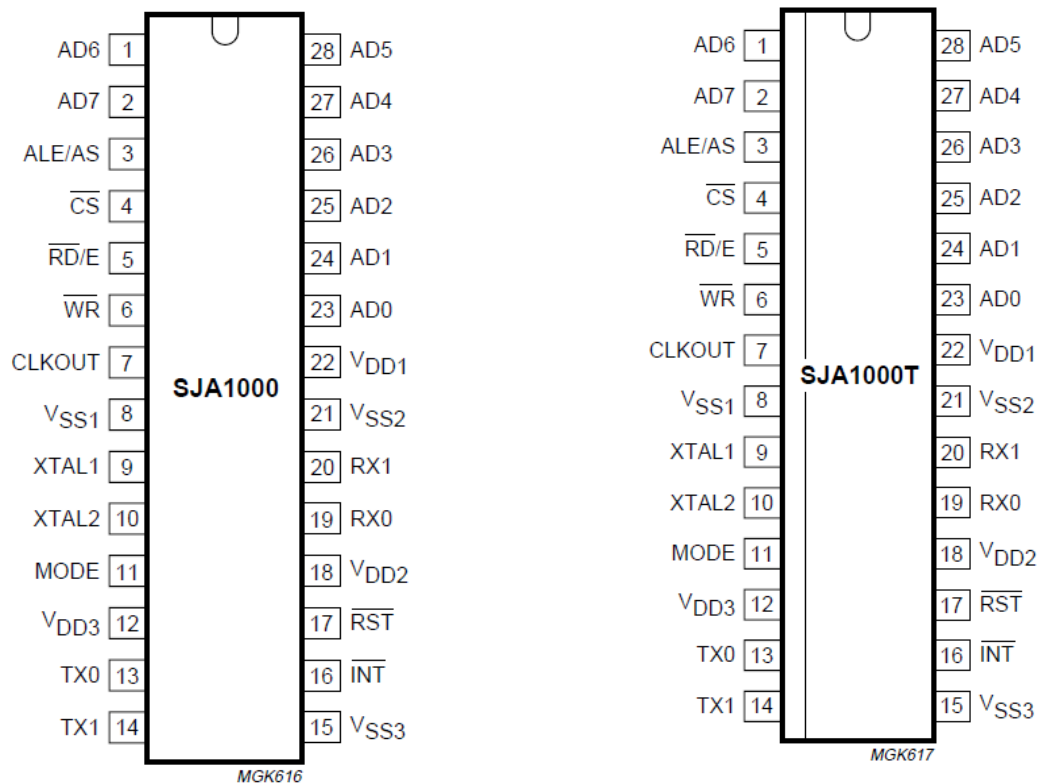
是新的操作模式。它能夠處理所有CAN2.0B規範的框架類型。而且它還提供一些增強功能，使SJA1000能應用於更寬的領域。工作模式藉由時鐘分頻暫存器中的CAN模式位元來選擇，重置時預設模式是Basic CAN模式。

SJA1000的應用



SJA1000 控制器引腳

- SJA1000 控制器有DIP28（圖左：塑質雙列直插封裝）和S028（圖右：塑質小型外線封裝）兩種形式



傳送緩衝器

- BasicCAN mode: 緩衝器有10-bytes.
 - 2 identifier bytes
 - up to 8 data bytes.

CAN ADDRESS	FIELD	NAME	BITS							
			7	6	5	4	3	2	1	0
10	descriptor	identifier byte 1	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5	ID.4	ID.3
11		identifier byte 2	ID.2	ID.1	ID.0	RTR	DLC.3	DLC.2	DLC.1	DLC.0
12	data	TX data 1	transmit data byte 1							
13		TX data 2	transmit data byte 2							
14		TX data 3	transmit data byte 3							
15		TX data 4	transmit data byte 4							
16		TX data 5	transmit data byte 5							
17		TX data 6	transmit data byte 6							
18		TX data 7	transmit data byte 7							
19		TX data 8	transmit data byte 8							

傳送緩衝器

- PeliCAN mode:緩衝器有 13 bytes.
 - 1 byte for Frame Information
 - 2 or 4 identifier bytes (Standard Frame or Extended Frame)
 - up to 8 data bytes.

標準frame 格式

CAN address	16	TX frame information
	17	TX identifier 1
	18	TX identifier 2
	19	TX data byte 1
	20	TX data byte 2
	21	TX data byte 3
	22	TX data byte 4
	23	TX data byte 5
	24	TX data byte 6
	25	TX data byte 7
	26	TX data byte 8
	27	unused
	28	unused

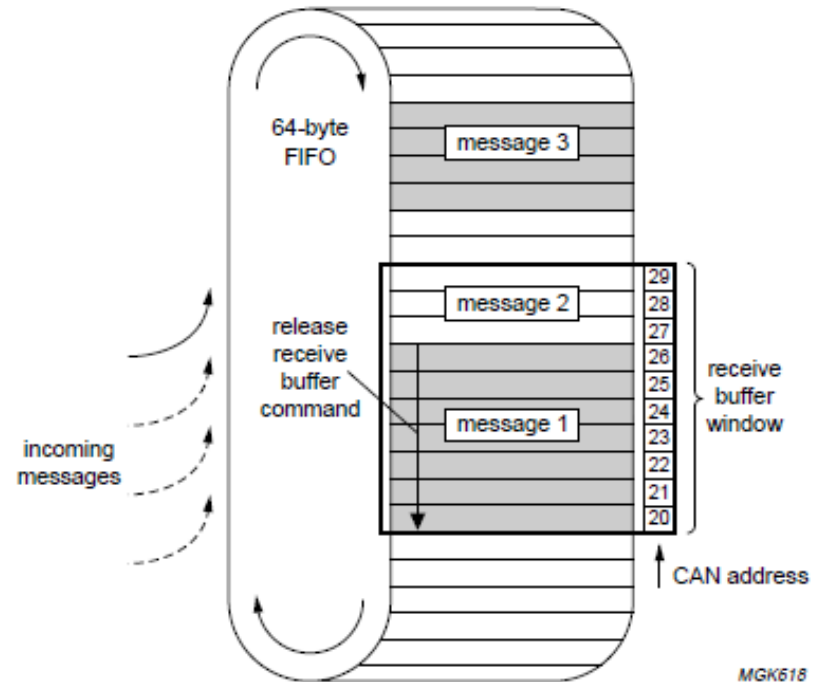
擴展frame格式

CAN address	16	TX frame information
	17	TX identifier 1
	18	TX identifier 2
	19	TX identifier 3
	20	TX identifier 4
	21	TX data byte 1
	22	TX data byte 2
	23	TX data byte 3
	24	TX data byte 4
	25	TX data byte 5
	26	TX data byte 6
	27	TX data byte 7
	28	TX data byte 8

MGK621

接收緩衝器RXFIFO

- RXFIFO總共有64個位元組的空間。
- 在任何時刻，可以被儲存在FIFO中的訊息數量決定於每個訊息的長度。
- 如果在RXFIFO沒有足夠的空間儲存新的消息，CAN控制器產生數據溢出，此消息被刪除。
- 這種情況藉由狀態暫存器和數據溢出中斷通知微控制器。



接收過濾器-BasicCAN

- 過濾器由2個8位元的暫存器控制:
 - Acceptance Code Register (ACR): 指定ID10-3的ID號碼，必須相同。
 - Acceptance Mask Register (AMR): 1表示任意值，0表示根據ACR的設定值。
- CAN訊息節點的8個最大有效位元和暫存器比較，決定是否接收。例如

	10	9	8	7	6	5	4	3	2	1	0
ACR	0	1	1	1	0	0	1	0			
AMR	0	0	1	1	1	0	0	0			
可接收的ID	0	1	x	x	x	0	1	0	x	x	x

接收過濾器-PeliCAN

- 過濾器由4個ACR和4個AMR暫存器控制:
Acceptance Code Register (ACR): 指定ID10-3的ID號碼，必須相同。
Acceptance Mask Register (AMR): 1表示任意值，0表示根據ACR的設定值。
- CAN訊息節點的8個最大有效位元和暫存器比較，決定是否接收。例如

	0	1(upper 4bits)	2	3
ACR	01XX-X010	XXXX-____	XXXX-XXXX	XXXX-XXXX
AMR	0011-1000	1111-____	1111-1111	1111-1111
可接收的ID	01XX-X010	XXXX-____		

串列傳輸速率設定

- 匯流排時脈暫存器0(BTR0)

BTR0定義鮑率預除值(BRP)和同步跳轉寬度(SJW)的值。在重置模式，這個暫存器可以讀/寫。在操作模式，PeliCAN只能讀，BasicCAN其值為FFH。

- 鮑率預除值(BRP)

CAN的時脈 t_{SCL} 的週期是可規劃的，以決定位元的時間。CAN時脈由如下公式計算：

$$t_{SCL} = 2 \times t_{CLK} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$$

其中 $t_{CLK} = XTAL$ 的頻率週期 = $1 / f_{XTAL}$

串列傳輸速率設定

- 同步跳轉寬度(SJW)
 1. 為了補償在不同匯流排控制器的時脈振盪器之間的相位偏移，任何匯流排控制器必須在目前傳送的相關信號邊緣重新同步。
 2. 同步跳轉寬度，定義了每一位元週期可以被重新同步，縮短或延長的最大時脈的數目：
 3. $t_{SJW} = t_{SCLX}(2 \times SJW.1 + SJW.0 + 1)$

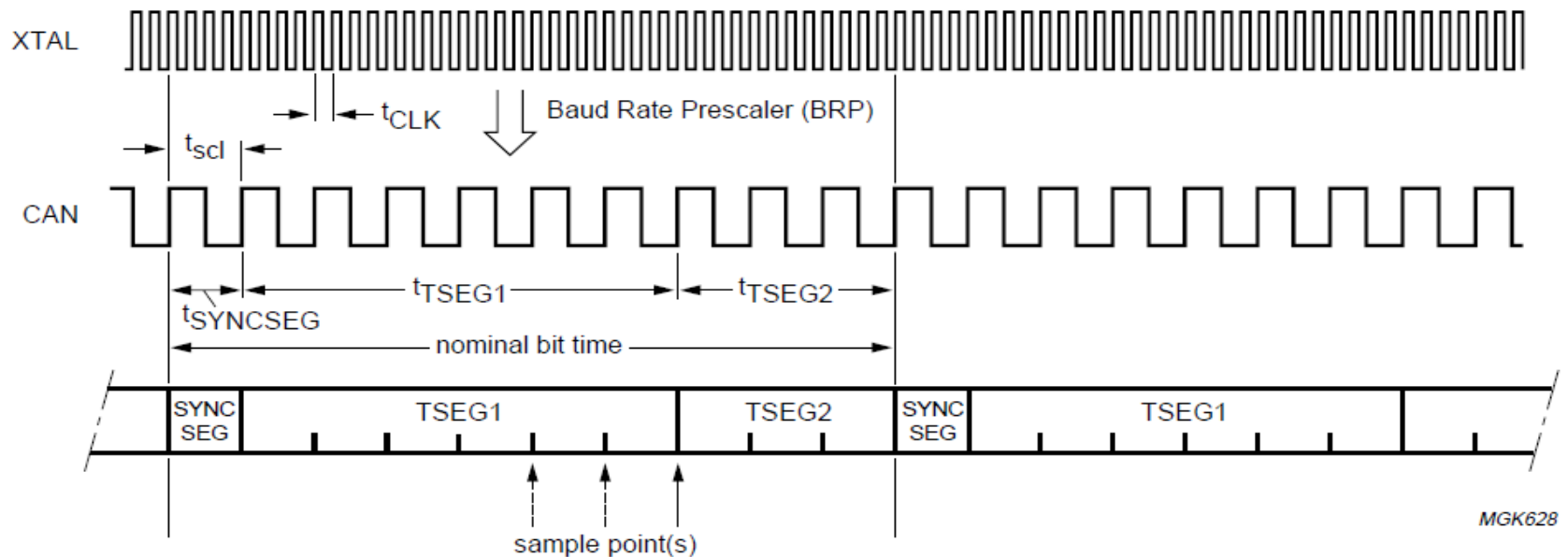
串列傳輸速率設定

- 匯流排時脈暫存器1(BTR1)
- 匯流排時脈暫存器1定義了每個位元週期的長度、取樣點的位置和在每個取樣點的取樣數目。
- 在重置模式中，這個暫存器可以被讀/寫。在操作模式，在PeLiCAN這個暫存器是唯讀的，在BasicCAN其值為FFH。
- SAM：=1時取樣3次；=0時取樣1次。

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SAM	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0

串列傳輸速率設定

- 時間段1 (TSEG1) 和時間段2 (TSEG2)
- $t_{\text{SYNCSEG}} = 1 \times t_{\text{SCL}}$
- $t_{\text{TSEG1}} = t_{\text{SCL}} \times (8 \times \text{TSEG1.3} + 4 \times \text{TSEG1.2} + 2 \times \text{TSEG1.1} + \text{TSEG1.0} + 1)$
- $t_{\text{TSEG2}} = t_{\text{SCL}} \times (4 \times \text{TSEG2.2} + 2 \times \text{TSEG2.1} + \text{TSEG2.0} + 1)$



建立CAN Bus通訊

- 系統開機後：
 - 設定主機和SJA1000相關的硬體和軟體
 - 設定SJA1000的模式、接收過濾器、位元時脈等
- 在主程式執行中：
 - 準備傳送的訊息和啟動SJA1000來傳送
 - 處理接收的訊息
 - 處理通訊中的錯誤

初始化

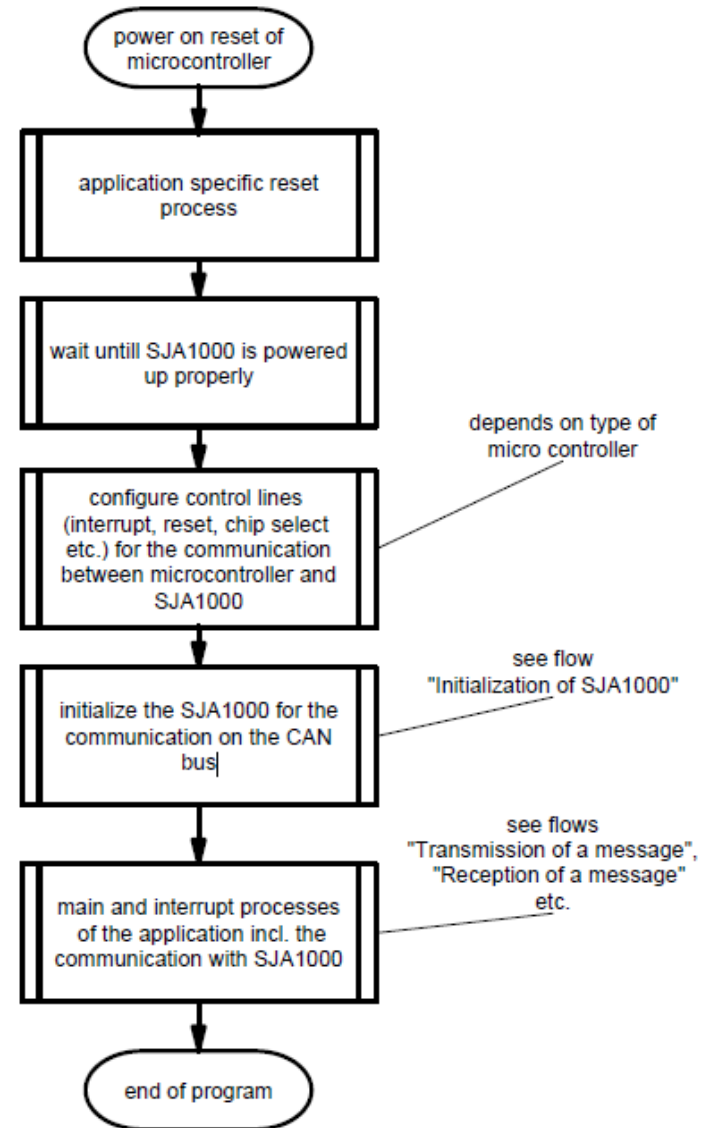
- 模式暫存器 (只有PeliCAN模式), 選擇下列的操作模式
 - 接收過濾器模式
 - 自我診斷模式
 - 只有傾聽模式
- 時脈除數暫存器, 定義
 - 如果使用BasicCAN或PeliCAN模式
 - 如果接腳CLKOUT致能可用
 - 如果繞過CAN輸入比較器
 - 如果TX1輸出用來作為接收中斷輸出

初始化

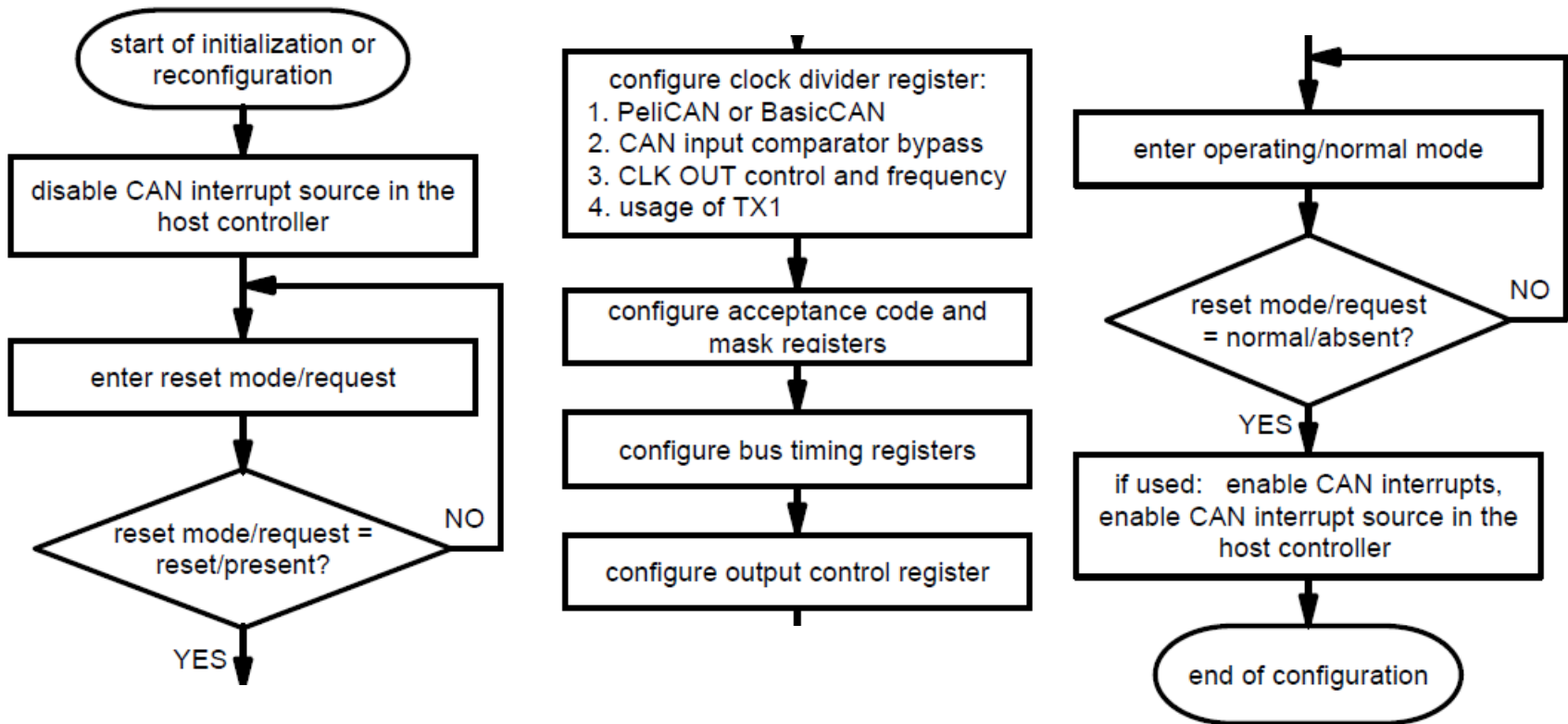
- ACR和AMR暫存器
 - ACR定義可接受的訊息，必須相同
 - AMR定義接受的訊息節點的遮罩，1表示此位元0/1都可以；0表示必須與ACR指定的相同。
- 匯流排時脈暫存器
 - 定義位元速率
 - 定義位元的取樣點
 - 定義取樣的次數
- 輸出控制暫存器
 - 定義接腳TX0和TX1輸出模式, 時脈輸出模式, 輸出輸出模式
 - 定義接腳TX0和TX1浮動, 下拉, 上拉或推/拉和極性

一般流程圖

- 重置設定
- 等待SJA1000啟動
- 設定主機和SJA1000之間的控制
- 初始化SJA1000
- 執行主程式、使用SJA1000傳送/接收資料

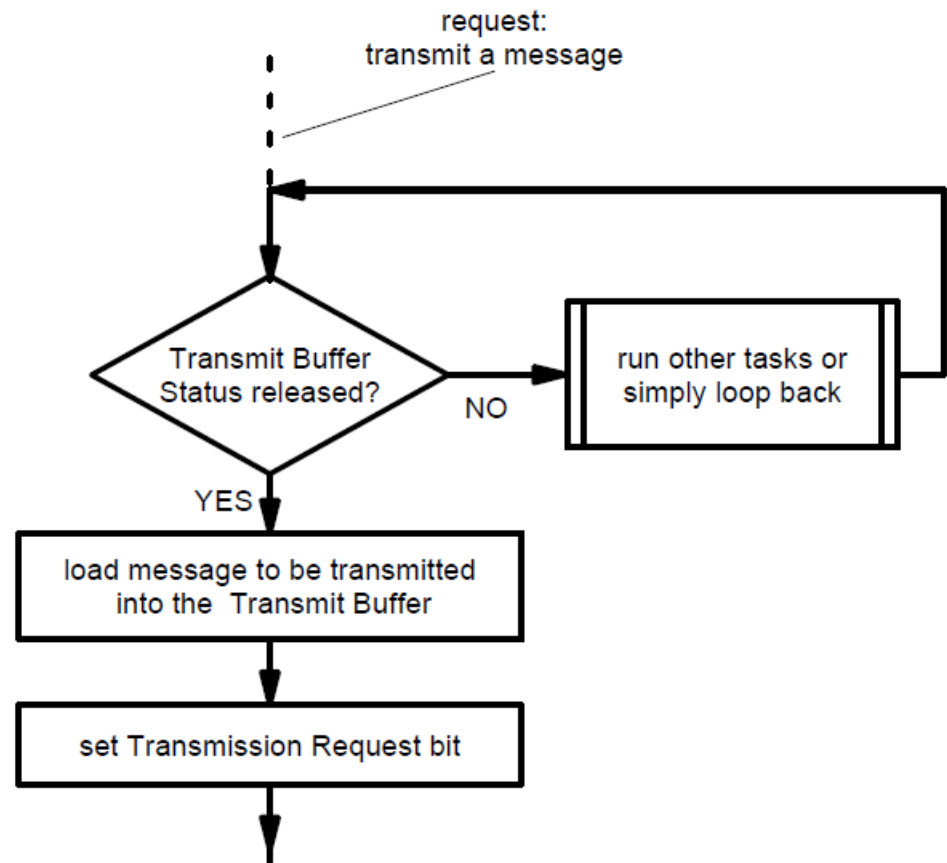


初始化



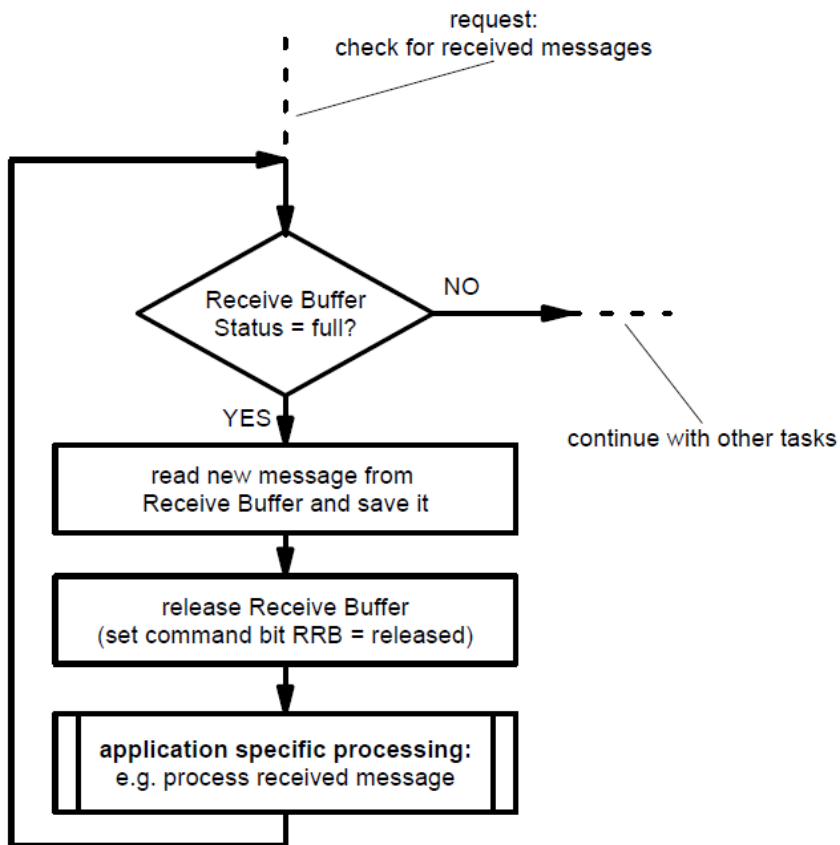
CAN匯流排資料發送(polling)

- 不使用中斷的方式
- 檢查[傳輸暫存器狀態 (TBS)]的旗標，直到傳輸暫存器釋放。
- 將新訊息寫入傳輸暫存器，設定[傳輸要求 (TR)]旗標，SJA1000開始傳送。

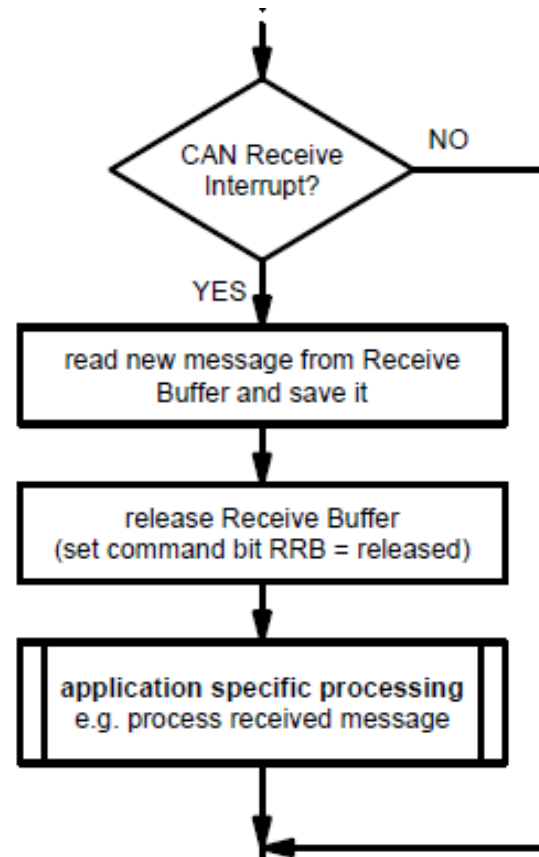


CAN匯流排資料接收

查詢控制接收資料流程圖



中斷控制接收資料流程圖



~ END ~