

• CAN / CANFD 總綫基本理論

Liquo berchiliqui at del idi omnis.

Tem aliquē

Vendis alis molenda vollamet arum et, quamus modi adionse quundaectia cus aut quias sedigni.

Idessi ut resende

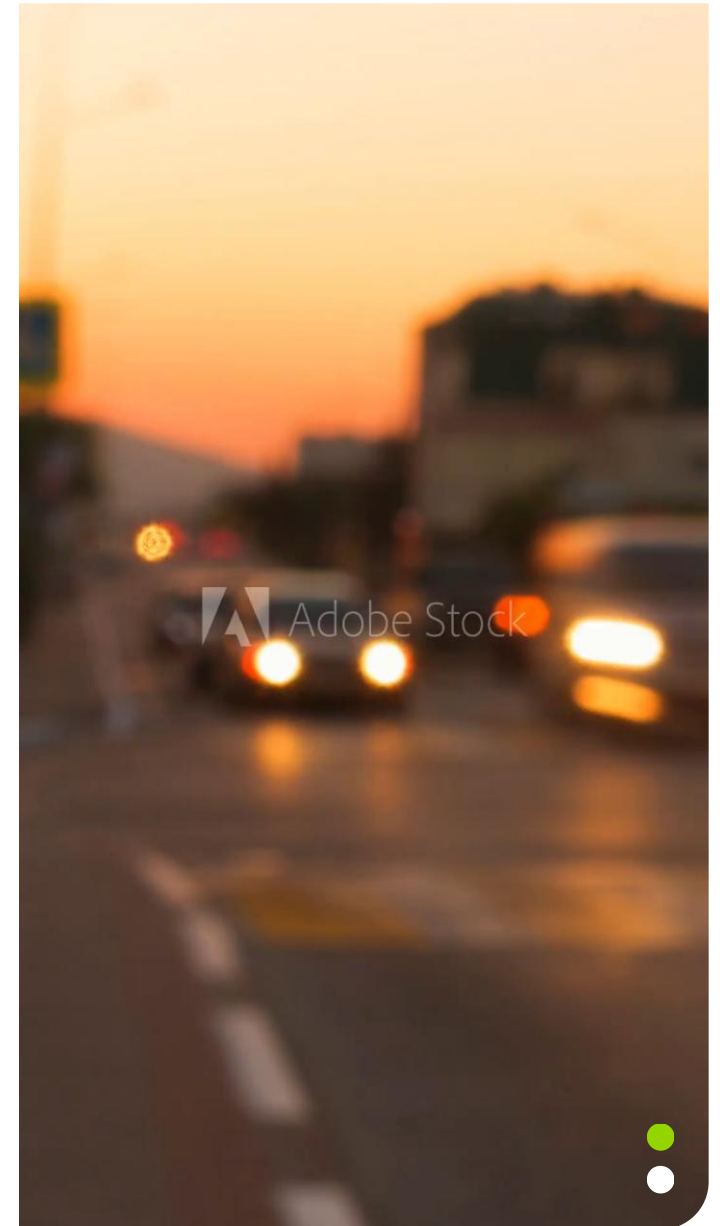
Bistrum quam doluptur rem etas dus dent que quibus quatus, occum raturum volupicidel endusdam dolorum.

Quatem

Invendae non pe namus, tem reicat dit ut reptati occus rest idellore non repreius.

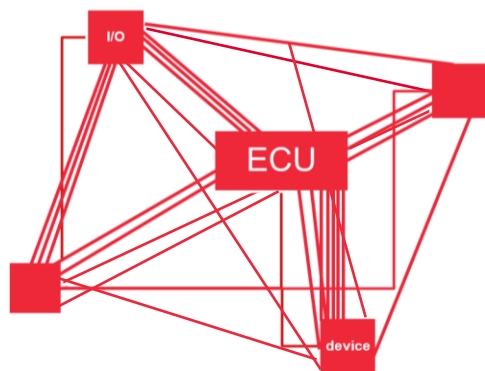
Corunt harciis

Eicaborrorit ea sus remquis dentur a vendaes eatiunt.

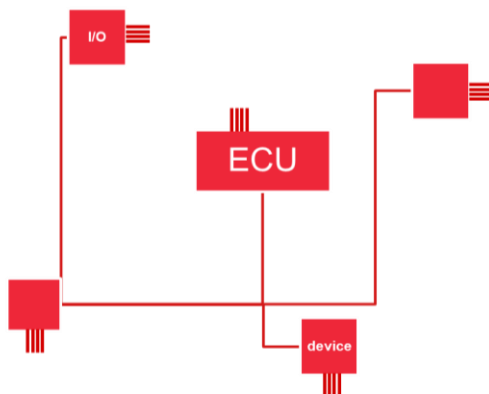


什麼是CAN總線？

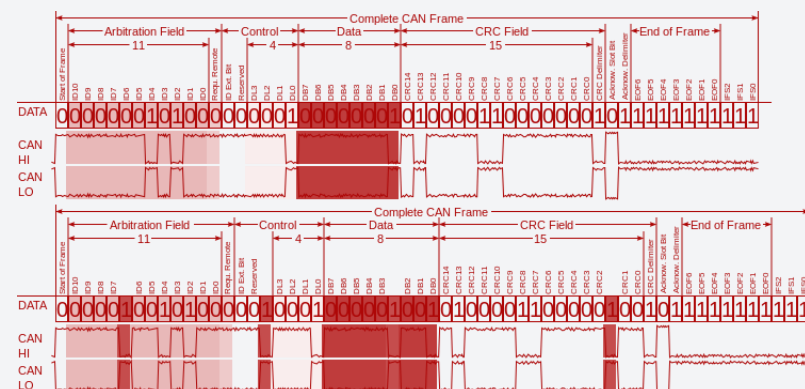
非CAN總綫



CAN總綫



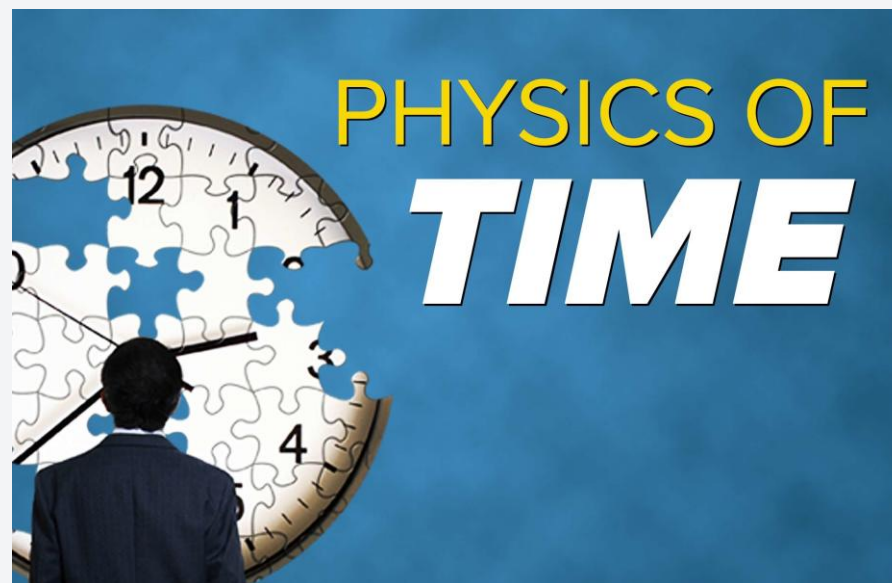
“ 控制器區域網路 (CAN 總綫) 是一種強大的嵌入式系統通信，旨在允許微控制器和設備在沒有主機的情況下與彼此的應用程式進行通訊。 ”



什麼是系統通信

通訊永遠是時間與物理之間的鬥爭

- 時間：控制訊號應在盡可能短的時間內發送。
 - 例如，盡可能快速地從 A 向 B 發送訊號。
- 物理：訊號完整性、無訊號遺失、幹擾
 - 例如，從 A 發送的內容與在 B 收到的內容完全一致



物理與時間

訊號衰減：電纜越長，訊號強度越弱。

電纜的傳播延遲。

黃色：起點 A

綠色：終點 B

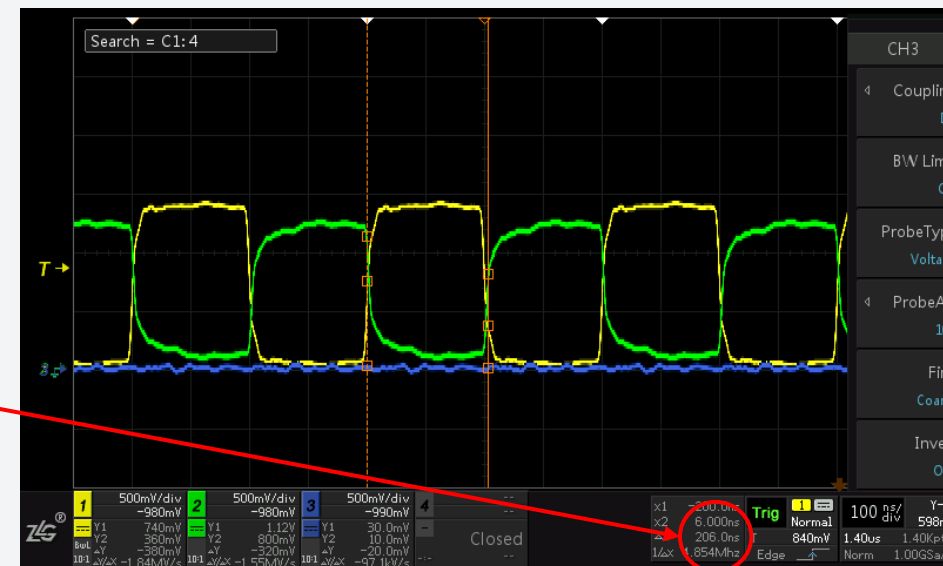
第一張圖：A 和 B 之間的距離 = 15 米

第二張圖：A 和 B 之間的距離 = 42 米

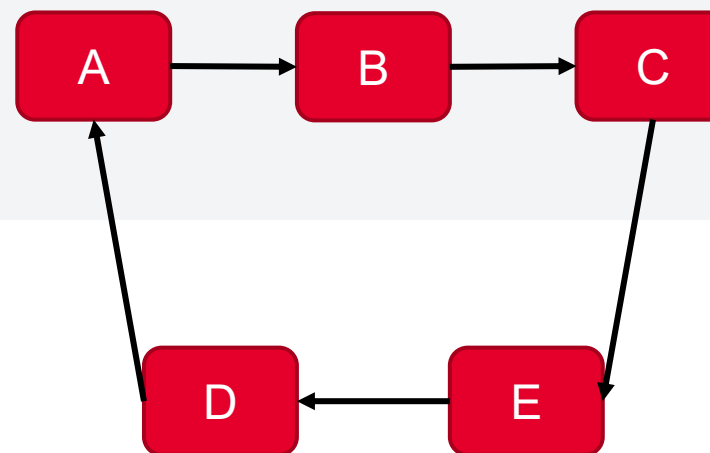
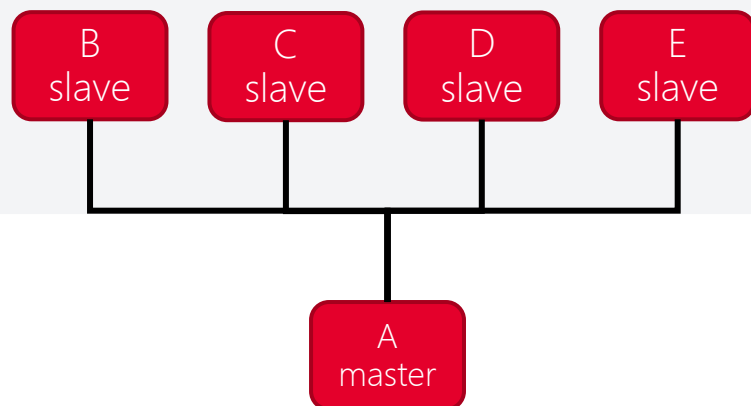
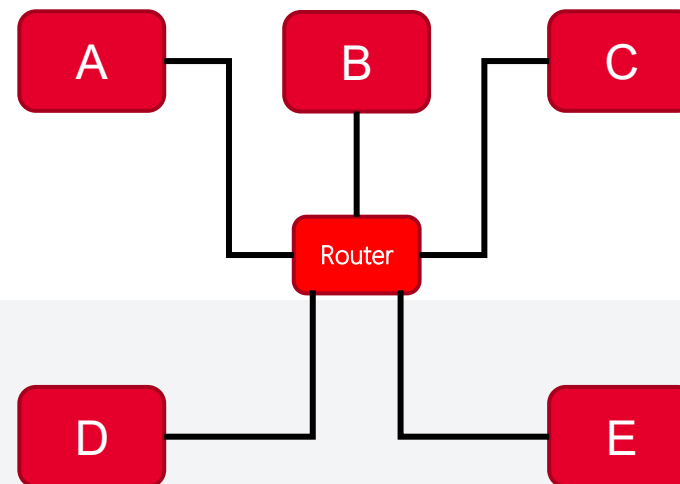
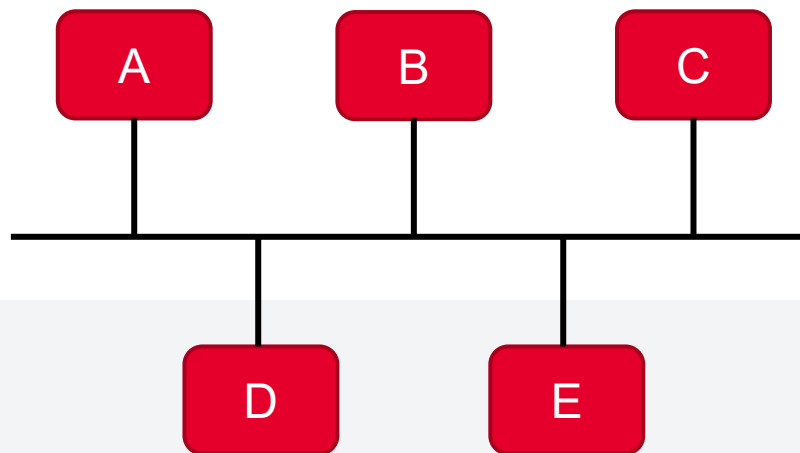
Travel Time between A and B = 62 nano second



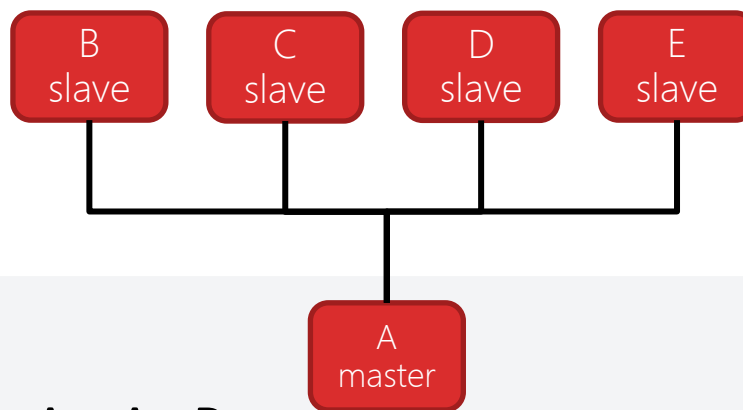
Travel Time between A and B = 206 nano second



常見的通信通訊



用戶端-伺服器（Client-Server）主從架構



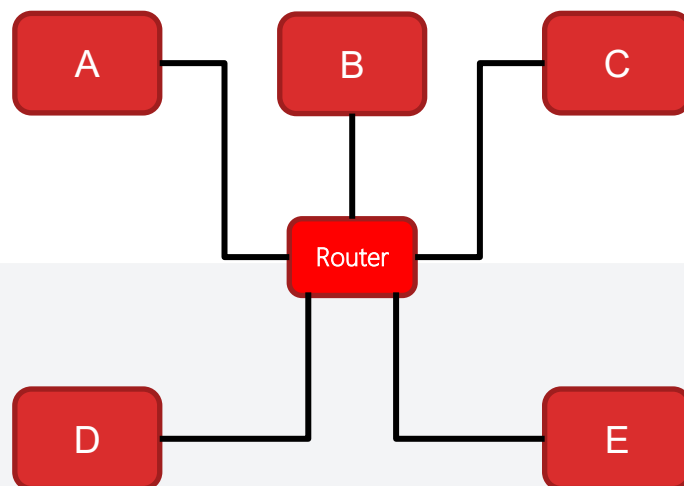
A->B, B->A, A->C, C->A, A->D...

BCDE 之間的通訊必須透過 A（主站）進行。

主站向從站發送請求，從站回覆訊息。

從站之間的通訊距離需要增加 2-3 倍才能進行資料交換。

路由系統架構



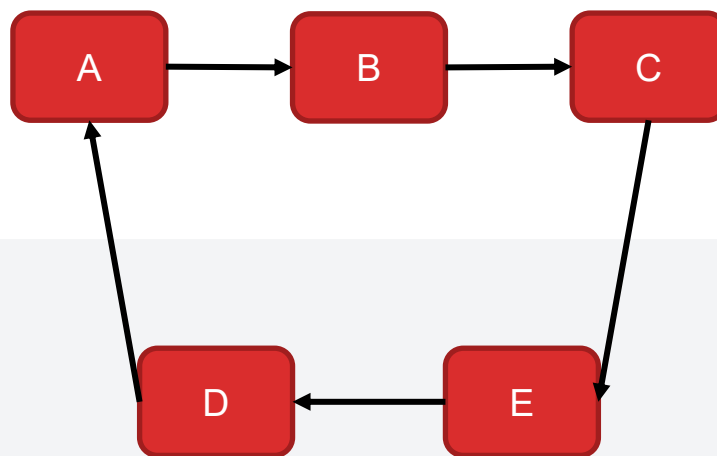
A->路由器->D, B->路由器->E...

路由器需要一定的時間來分析和處理節點之間的路由資訊。

比喻：直飛還是中途停留？

如果路由器出現問題，整個網路就會癱瘓。

迴路控制系統



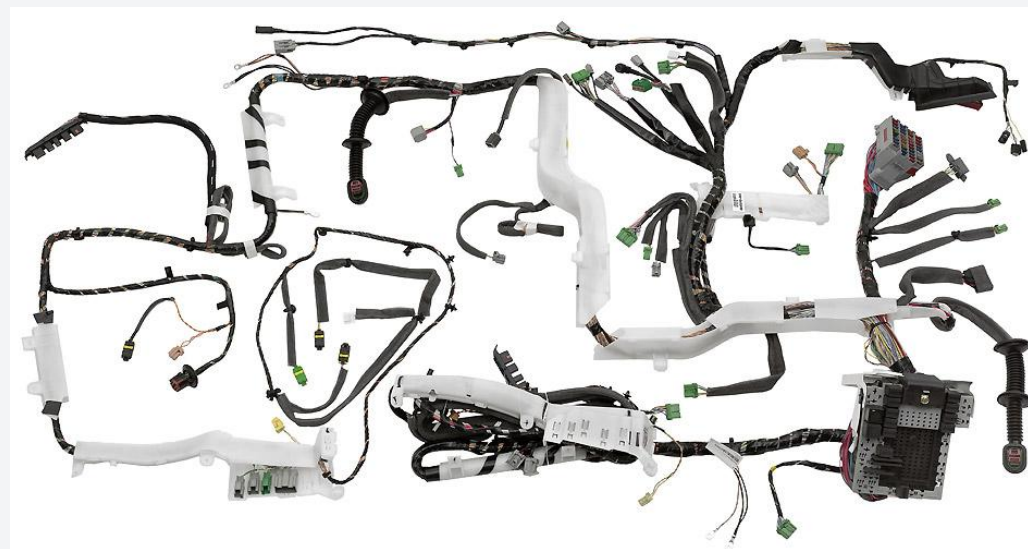
A->B->C->D->E->A

回饋迴路控制，每個節點的訊息將持續發送，直到整個往返行程完成。

長距離時，A 必須等待工作週期完成才能發送下一個控制訊息/訊號。

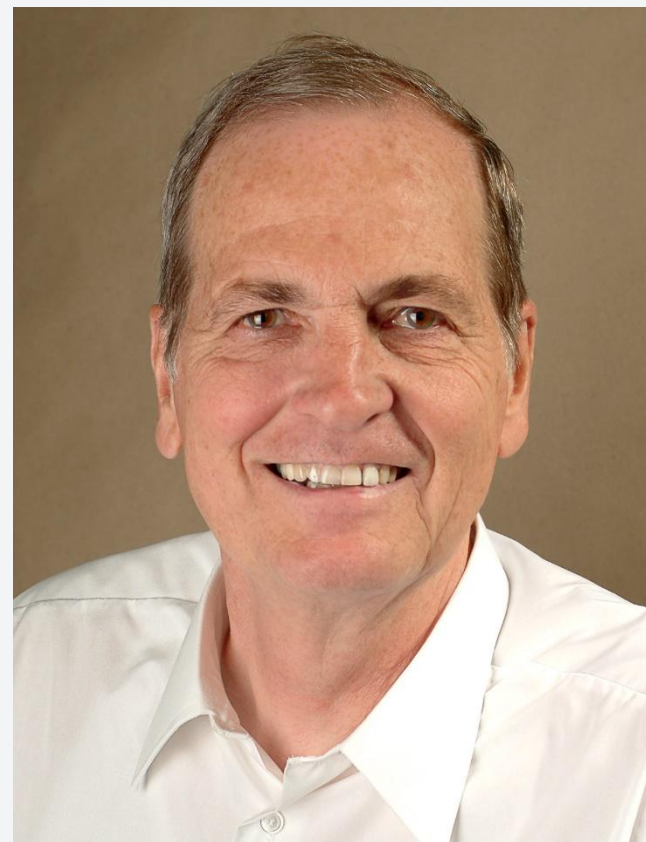
系統通訊最常遇見的問題

- 傳播延遲：速度受電纜長度限制
- 訊號傳播路徑
- 訊號傳輸與接收模型：點對點、廣播、路由、主從站
- 通訊衝突：如何處理兩個或多個節點同時發送數據
- 在通訊過程中，如果物理層出現問題，例如乾擾、校驗和錯誤、阻抗不一致、速度不正確等
- 實體網路可靠性，例如網路上的節點故障

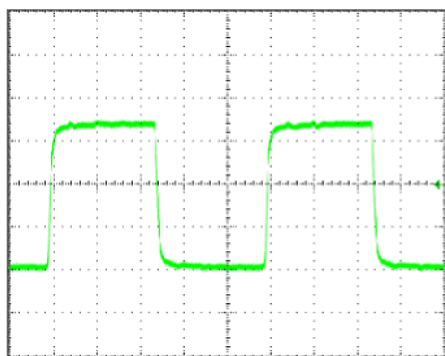
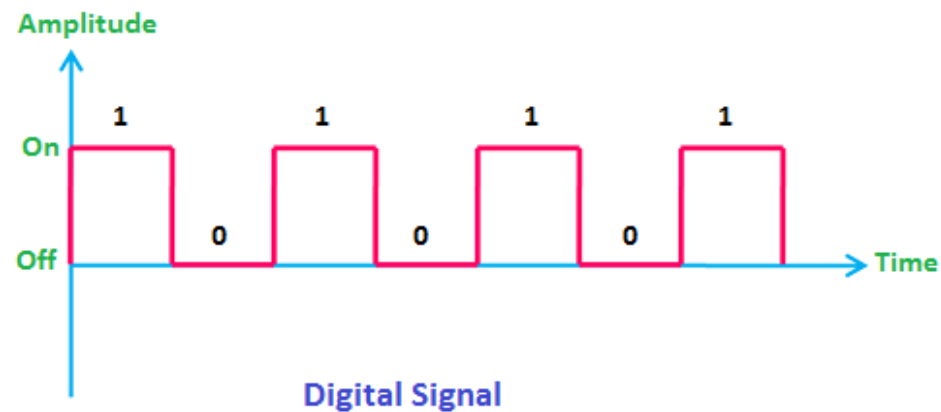


UWE KIENCKE

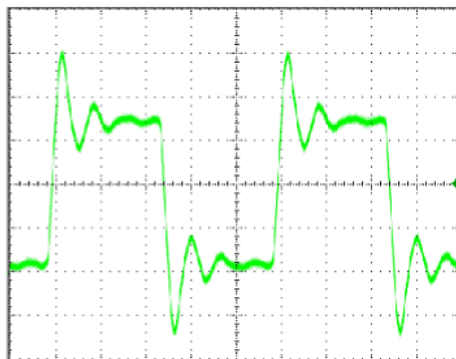
- CAN 總綫之父
- 1983年開始研發新的串行總綫系統
- CAN（ISO-11898）擁有近40年的傳奇歷史
- 完全適用於即時控制系統、分佈嵌入式系統
- 關鍵字：廣播、即時、仲裁、可靠性、低成本、差分訊號、抗干擾、靈活的總綫拓撲



數字訊號的世界

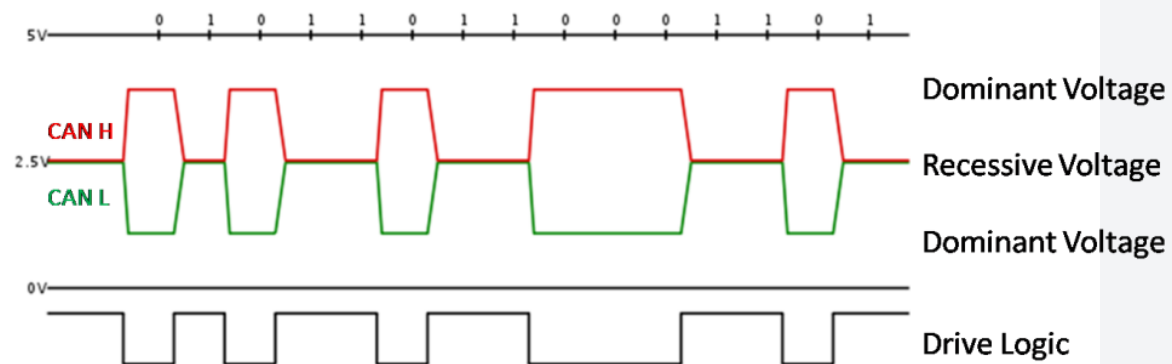
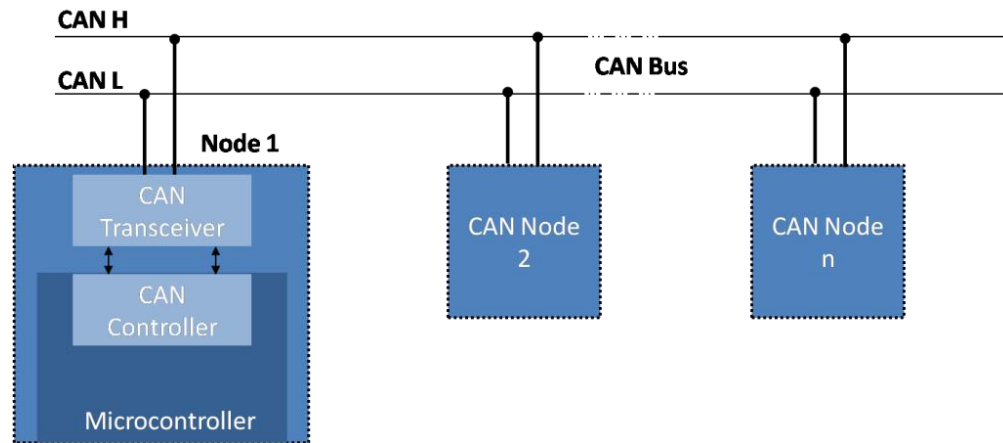


a.



b.

理想與現實的區別



CAN H 和 CAN L 為通訊訊號提供保障

CAN H – CAN L = 數字 0 或 1 邏輯

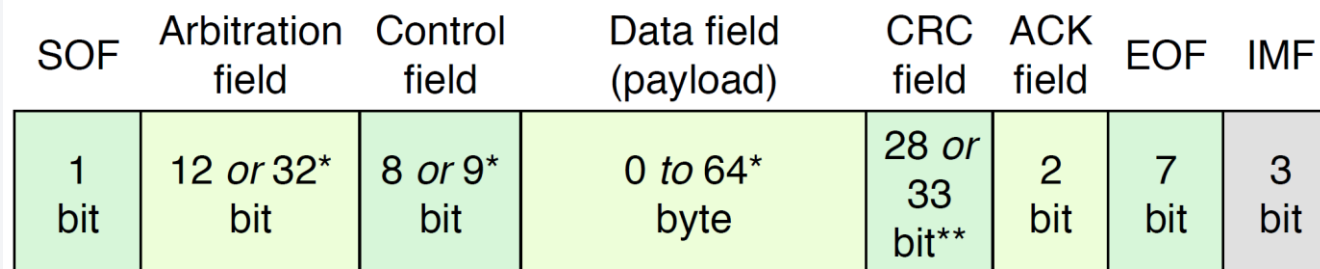
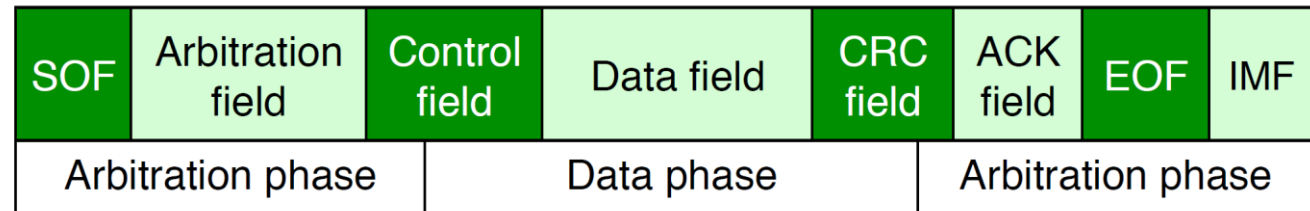
(顯性位) $3.5V - 1.5V = 2V$

(隱性位) $2.5V - 2.5V = 0V$

	Low speed CAN		High speed CAN	
	Dominant	Recessive	Dominant	Recessive
$V_{CAN\ H}$	3.6 V	0 V	3.5 V	2.5 V
$V_{CAN\ L}$	1.4 V	5 V	1.5 V	2.5 V

		CAN
1	網絡拓樸	總綫
2	傳播媒體	雙絞綫
3	有效總綫長度	5 km to 40 m
4	分支節點長度	30 cm (建議)
5	速率	10 kbps to 1 Mbps(CAN FD 8Mbps)
6	最短幀長度	47 bit (47 us at 1Mbps)
7	最長幀長度	160 bit (.16 ms at 1Mbps)
8	訊息衝突無損機制	有
9	最大訊息衝突延遲	最高優先權訊息可預測 (1 Mbps 時為 160 us)
10	尋址方式	內容尋址
11	重傳機制	有
12	錯誤時的重大重傳延遲	187 位元用於最高優先級別訊息
13	保證數據一致性	有

CAN 報文結構



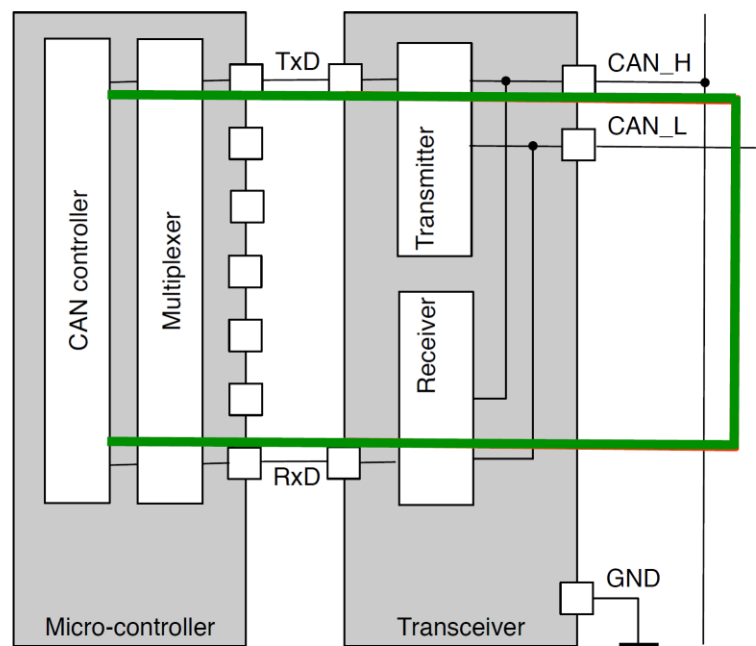
CAN ID
標識符

資料數據荷載：資訊、
狀態、控制訊號...等

CAN 總綫協議 – 如何確保通訊完整

1. 位元監控 Bit Monitoring

CAN總綫上的驅動都會讀回已發送的訊號電平。如果實際讀取的位元電平與發送的位元電平不同，則會發出位元錯誤訊號。



CAN 總綫協議 – 如何確保通訊完整

2. 位元填充 Bit Stuffing

當節點發送了五個連續的相同電平位元時，它會將第六個相反電平的位元加到輸出位元流中。

這樣做是為了避免在總線上出現過多的直流分量。

這也使得接收端擁有額外的檢查機制，並重新同步時鐘。

CAN 總綫協議 – 如何確保通訊完整

3. 幀檢查 Frame Check

CAN 封包中有一些固定的位元格式。

這明確定義了訊號到達這些位元時必鬚發生的電平變化。（這些部分包括 CRC 分隔符號、ACK 分隔符號、訊框結束符以及間歇符。）

如果 CAN 控制器偵測到這些固定欄位中的某個無效值，則會發出「格式錯誤」訊號。

SOF	Arbitration field	Control field	Data field	CRC field	ACK field	EOF	IMF
Arbitration phase		Data phase			Arbitration phase		
SOF	Arbitration field	Control field	Data field (payload)	CRC field	ACK field	EOF	IMF
1 bit	12 or 32* bit	8 or 9* bit	0 to 64* byte	28 or 33 bit**	2 bit	7 bit	3 bit

CAN 總綫協議 – 如何確保通訊完整

4. 應答位檢查 Acknowledgement Check

總線上所有正確接收訊息的節點都應在訊息中所謂的確認時隙 (ACK Slot) 中發送顯性電平。

發送器將在此處發送隱性電平。如果發送器無法在確認時隙 (ACK Slot) 中偵測到顯性電平，則會發出確認錯誤訊號。

SOF	Arbitration field	Control field	Data field	CRC field	ACK field	EOF	IMF
Arbitration phase		Data phase			Arbitration phase		
SOF	Arbitration field	Control field	Data field (payload)	CRC field	ACK field	EOF	IMF
1 bit	12 or 32* bit	8 or 9* bit	0 to 64* byte	28 or 33 bit**	2 bit	7 bit	3 bit

CAN 總綫協議 – 如何確保通訊完整

5.CRC 校驗

每個訊息都具有循環冗餘校驗和 (CRC)，並且任何節點偵測到訊息中的 CRC 與其自身計算的 CRC 不同，都會發出 CRC 錯誤訊號。

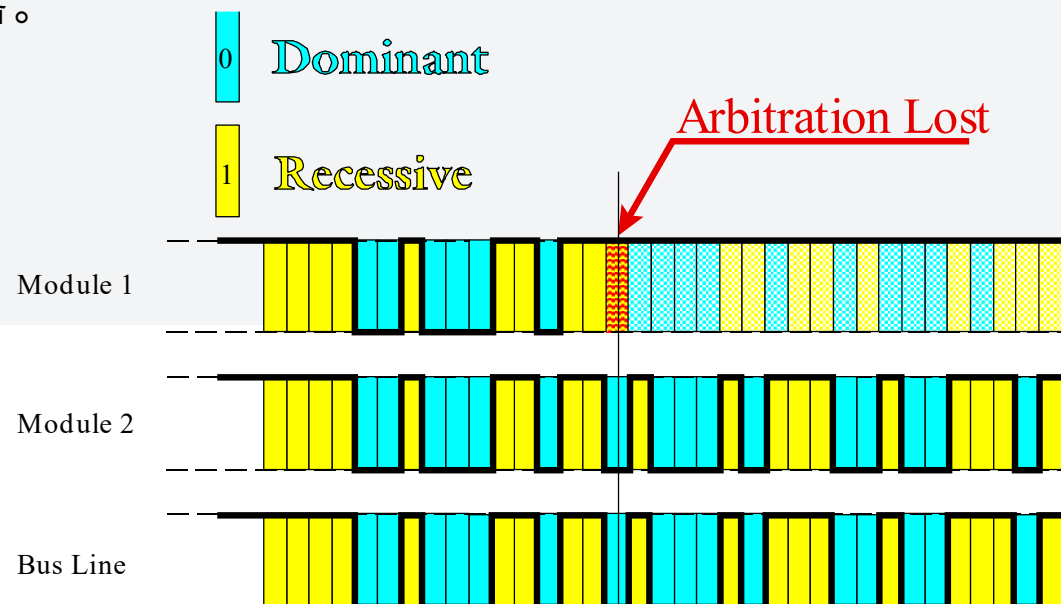
SOF	Arbitration field	Control field	Data field	CRC field	ACK field	EOF	IMF
Arbitration phase		Data phase			Arbitration phase		
SOF	Arbitration field	Control field	Data field (payload)	CRC field	ACK field	EOF	IMF
1 bit	12 or 32* bit	8 or 9* bit	0 to 64* byte	28 or 33 bit**	2 bit	7 bit	3 bit

CAN 總綫協議 – 如何確保通訊完整

6. 按位仲裁

每個 CAN 控制器都具有自我檢測機制（1. 位元監控）。

如果發送器發現一個自身未發送的顯性位，它將失去仲裁並繼續作為接收器。



CAN 總綫協議 – 如何確保通訊完整

6. 按位仲裁

理論 - 低壓電平拉低高壓電平

例子:

4 CAN ID 1,2,3,4

ID 1 = 001

ID 2 = 010

ID 3 = 011

ID 4 = 100

ID 4 將失去第一個仲裁。

ID 1、2、3 中第一位的顯性位元（邏輯 0）已將 ID 4 的第一個隱性位元（邏輯 1）拉低。

因此，當 ID 4 進行位元監控時，它發送的是邏輯 1，但接收的是邏輯 0。

在第二位，ID2 和 ID 3 將根據相同的原理失去對 ID 1 的仲裁。

時間軸



CAN 的優勢

- 從頭到尾進行訊號（位元）檢查
- 確保每一位的準確性和完整性。
- 更好的時間同步
- 提供資料一致性和完整性
- 時間和延遲可預測
- 在CAN中，沒有什麼是單一位元無法解決的。一切都由晶片內部的底層控制和檢查。

想想看，如果我們用軟體實現這些功能，需要多少行程式碼？

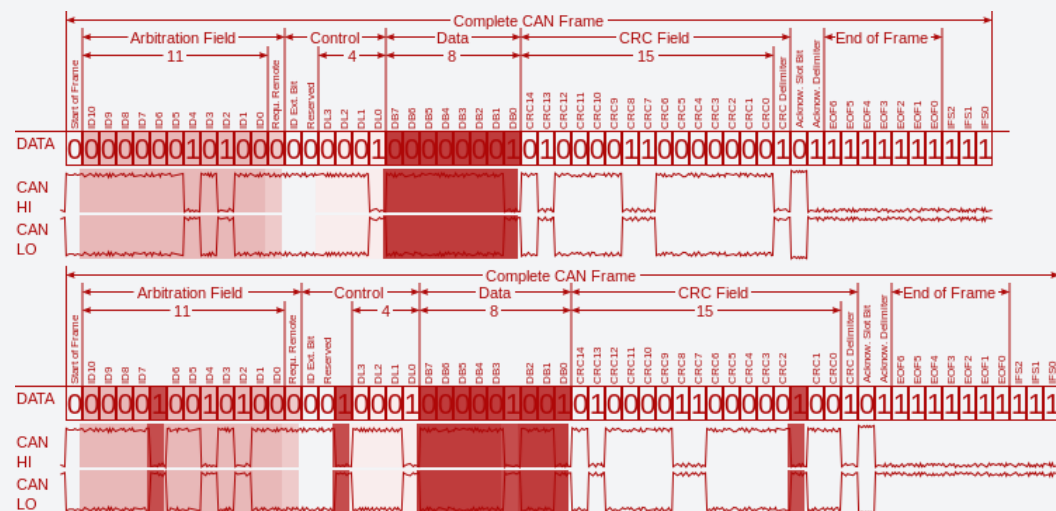
時間軸



CAN的優勢

- 從頭到尾進行訊號（位元）檢查
- 確保每一位的準確性和完整性。
- 更好的時間同步
- 提供資料一致性和完整性
- 時間和延遲可預測
- 在CAN中，沒有什麼是單一位元無法解決的。
一切都由晶片內部的底層控制和檢查。

想想看，如果我們用軟體實現這些功能，需要多少行程式碼？

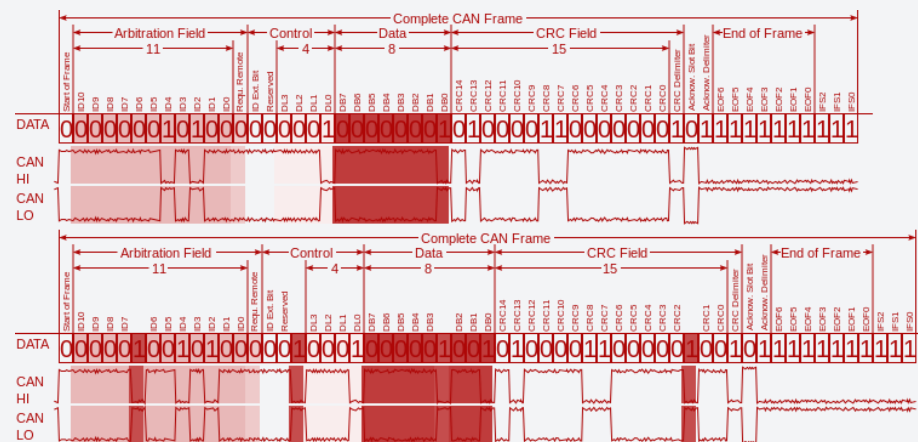


CAN的現狀

- CAN 總線有效波特率受限於：
- 傳播延遲
- 仲裁（CAN 總線採用廣播形式，在仲裁階段所有模塊同時發送）
- 它是 40 年前基於當時的技術定義的。（如今半導體技術已經快速發展並成熟）
- 其他通訊協定可以達到更高的波特率（速度），因為它們大多是點對點通訊。需要額外的專用連接器、線纜、路由器等，這意味著成本的增加。

思考問題：

- 1M CAN 和 10M 乙太網路哪個比較快?
- 處理衝突最有效的方法是什麼?
- 如何保證資料的一致性和完整性?

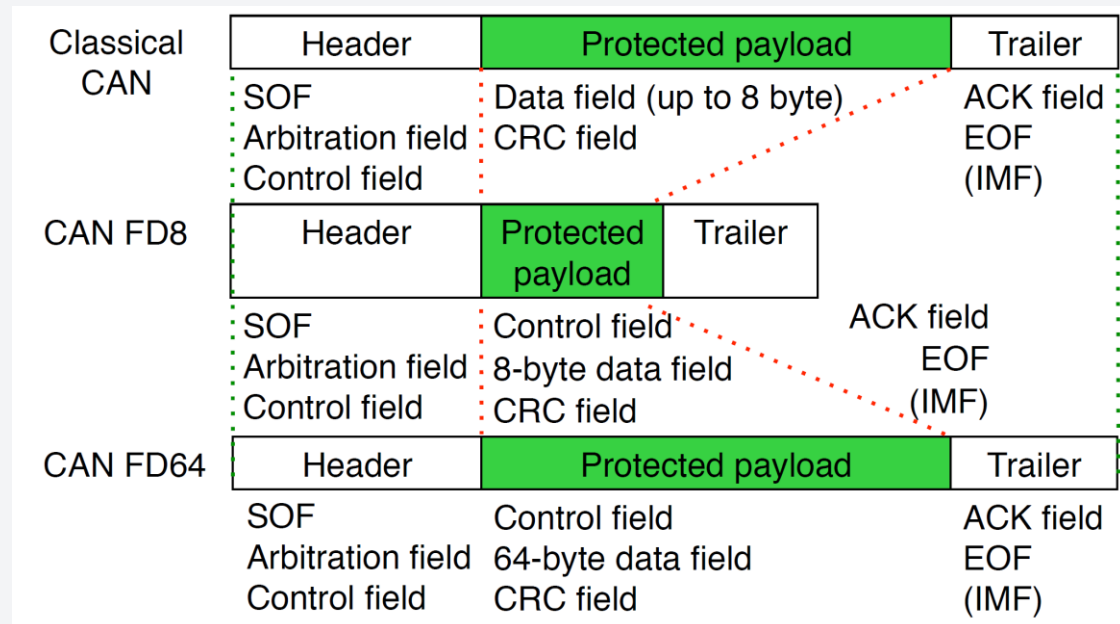


CAN FD的概念

- 整車系統功能逐漸強大
 - 需要更高的數據帶寬, 例如 功能安全, 網絡安全, 輔助駕駛功能
 - 系統的ECU數量也隨著功能性的增加而遞增, 更複雜的診斷系統, 刷寫...等
- 速度受限於仲裁部分, 仲裁後只有一個模組在傳輸, 速度不受限制。
 - 數據效率(Overhead)
 - 8位元組的資料不夠用。
 - 這是1985年的協議。
 - 如今的技術比35年前有了很大的進步, 例如CPU、記憶體、儲存等等。

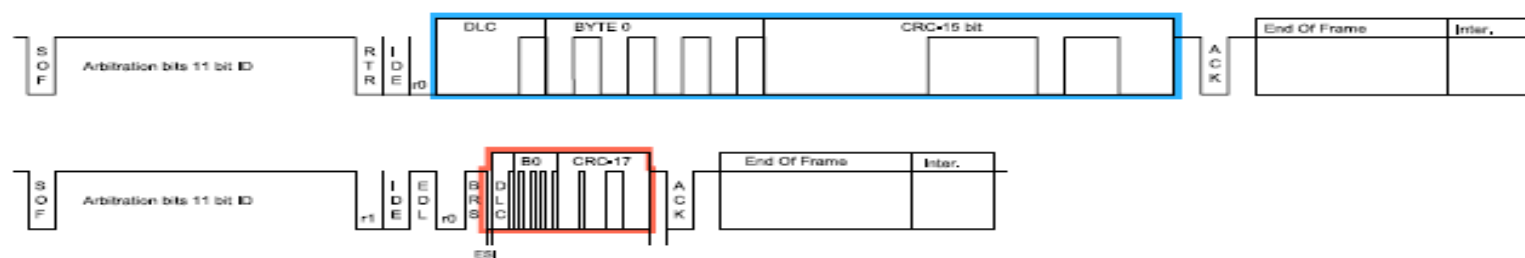
CAN總綫如何實現速率與數據的提升

- 仲裁欄位和確認欄位保持原有位元率。加速以下段落：
- DLC (數據長度)
- Data
- CRC 校驗
- 允許超過 8 個Byte的數據
- CAN FD 最多支援 64 個Bytes



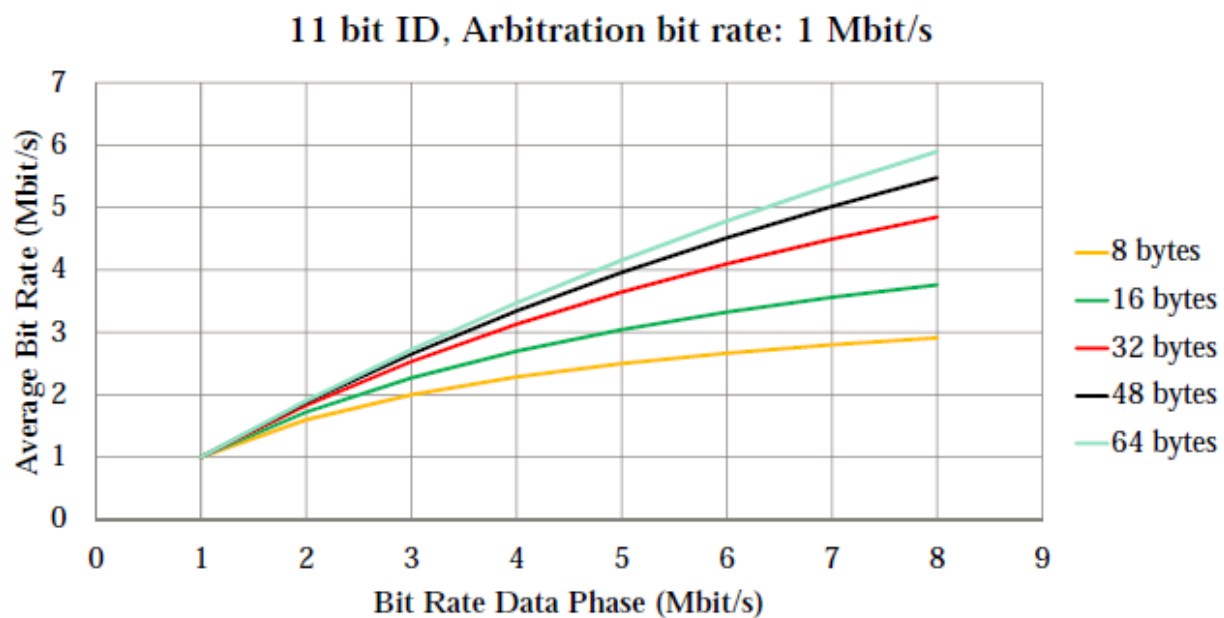
CAN 與CAN FD的比較

CAN
CAN FD

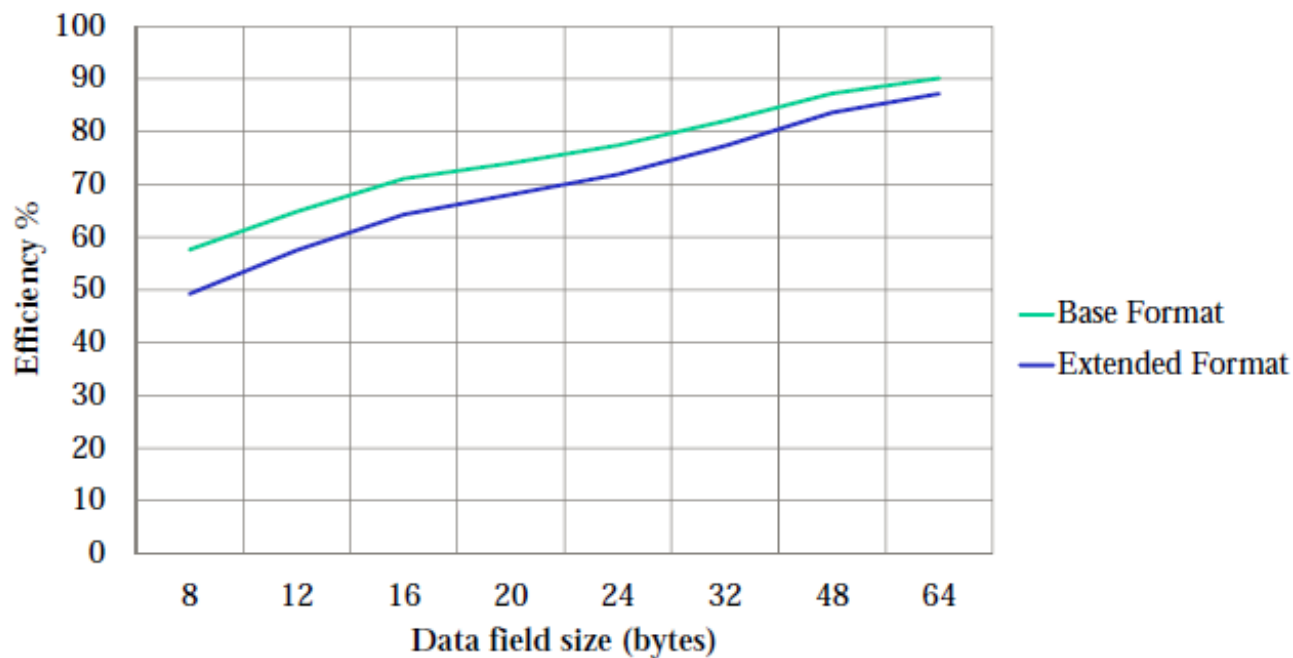


- CAN 報文發送1Byte數據
- CAN FD 發送相同數據, 數據以5倍速率提升

CAN FD 平均速率



CAN FD數據有效率



- 標準幀 ID 由 8 byte 58%, 提升到 64 byte 90%
- 擴展幀 ID 由 8 byte 49%, 提升到 64 byte 86%



Advancing connectivity

CONNECTING REAL-WORLD DATA TO A BRIGHTER FUTURE



TAIWAN PARTNER

SYSTEM ACCESS 集研科技

Kvaser 台灣技術支援與採購服務



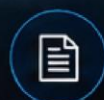
產品諮詢

Product Consultation



技術支援

Technical Support



選型建議

Product Selection



採購服務

Purchase Service

產品資料下載
Product Information



聯絡我們 / 詢價
Contact Us

