

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN ĐIỆN



BÀI GIẢNG

HỆ THỐNG ĐO VÀ ĐIỀU KHIỂN CÔNG NGHIỆP

Nguyễn Thị Hué

Bộ môn Kỹ thuật đo và Tin học công nghiệp

NỘI DUNG MÔN HỌC

1

Giới thiệu chung và lịch sử phát triển của các hệ thống đo và điều khiển công nghiệp

2

Các thiết bị đo lường và chấp hành trong công nghiệp

3

Các bộ điều khiển khả trình

4

Các thiết bị giám sát trong công nghiệp

5

Cơ sở kỹ thuật truyền tin công nghiệp

6

Các giao thức công nghiệp tiêu biểu

7

Một số hệ thống đo và điều khiển công nghiệp tiêu biểu

Tài liệu tham khảo

- Giáo trình “Màng thông tin công nghiệp” Hoàng Minh Sơn, nhà xuất bản Khoa học và Kỹ thuật
- Giáo trình “Hệ thống thông tin công nghiệp” Phạm Thượng Hàn (chủ biên) Nhà xuất bản giáo dục
- Giáo trình “Cảm biến công nghiệp”
- Bài giảng “Đo và điều khiển công nghiệp” Đào Đức Thịnh, bộ môn Kỹ thuật đo và Tin học công nghiệp.
-

NỘI DUNG CHƯƠNG 6

6

Các giao thức công nghiệp tiêu biểu

1. *MODBUS*

2. *AS-I*

3. *PROFIBUS*

4. *CAN*

5. *DEVICE NET*

6. *INTERBUS*

7. *FOUNDATION FIELDBUS*

8. *ETHERNET*

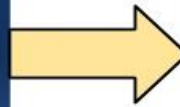
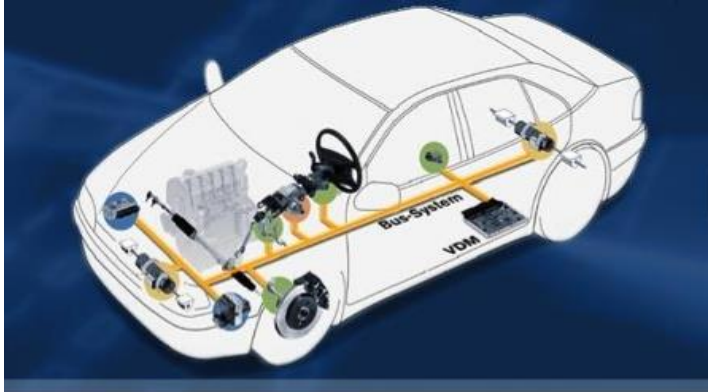
9. *HART*

CAN (Controller Area Network)

1. Lịch sử phát triển
2. Kiến trúc giao thức
3. Cấu trúc mạng và kỹ thuật truyền dẫn
4. Cơ chế giao tiếp
5. Điều khiển truy nhập bus
6. Mã hóa dữ liệu
7. Cấu trúc bức điện
8. Bảo toàn dữ liệu

1. Lịch sử phát triển

- Xuất phát là phát triển chung của 2 hãng Bosch và Intel
- Được chuẩn hóa quốc tế trong ISO 11898
- Nhờ một số ưu thế mà đã thâm nhập vào công nghiệp.



1. Lịch sử phát triển

- Thay thế cách nối điểm-điểm trong phương tiện giao thông cơ giới.

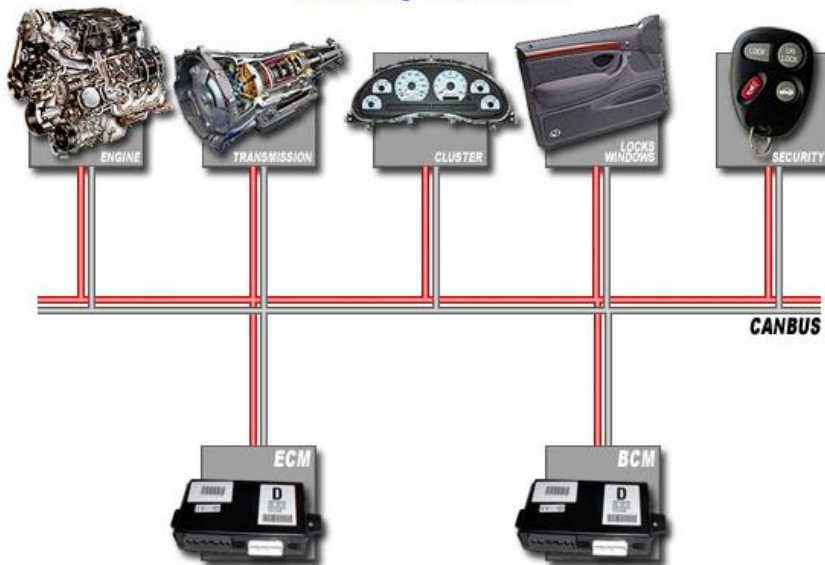
Nối dây truyền thống

Vehicle Wiring: conventional multi-wire looms



Nối dây theo CAN

Vehicle Wiring: CAN Bus network

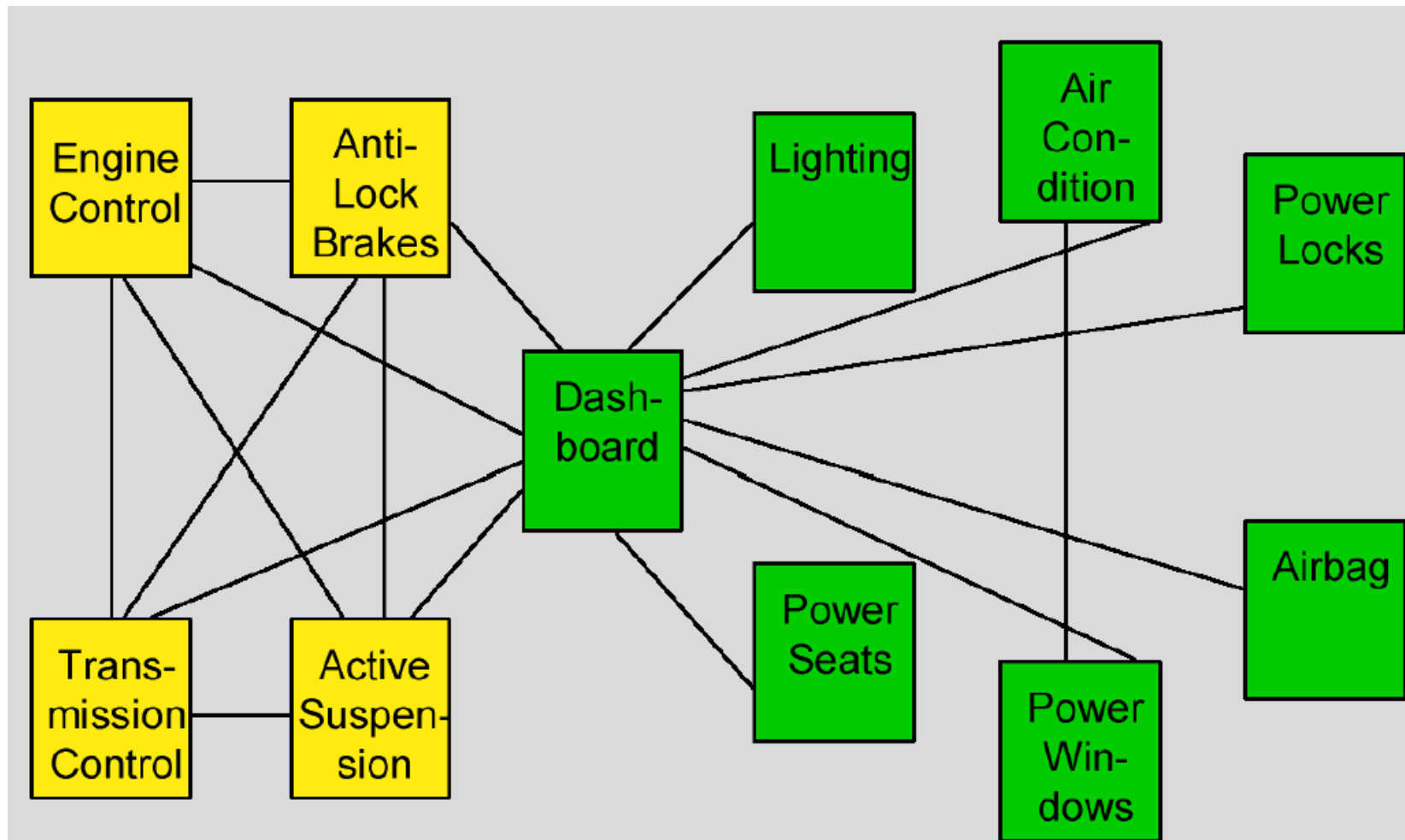


1. Lịch sử phát triển

- First introduced in February of 1986 by Robert Bosch GmbH
- Developed because existing serial buses in the early 1980s were not able to fulfill all the requirements to be used in passenger cars
- Intel released the first CAN controller chip in 1987
- In November 1993 the CAN ISO11898 standard was published
- First applications included use by an elevator manufacturer and some textile machine manufacturers
- Multiple higher level protocols for CAN have been developed since 1994

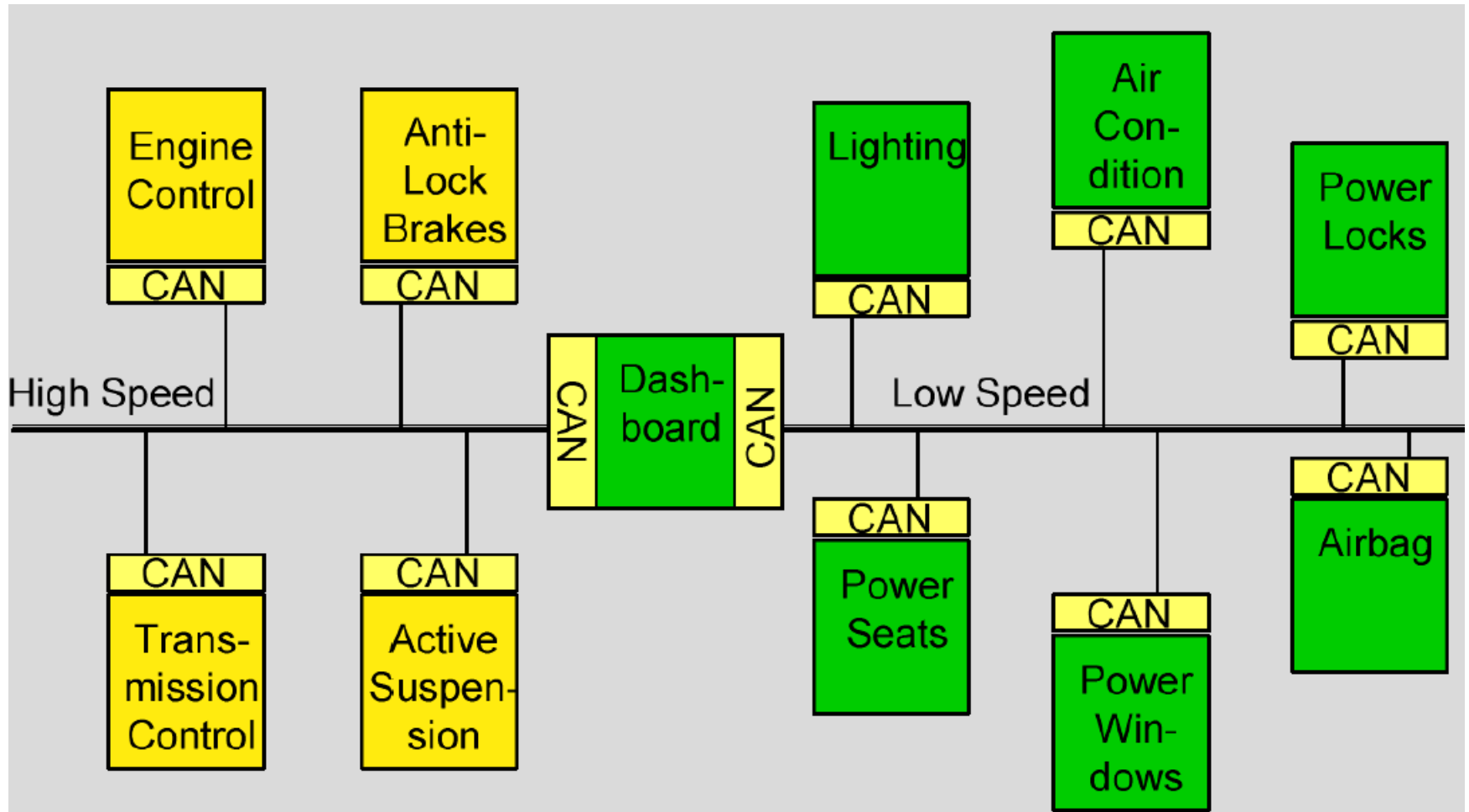
Before CAN

- Hệ thống cũ kết nối dây điểm điểm, sơ đồ nối phức tạp, khó quản lý



With CAN

- Kết nối điểm điểm thay thế bởi hệ thống kết nối thông qua hệ thống bus nối tiếp.

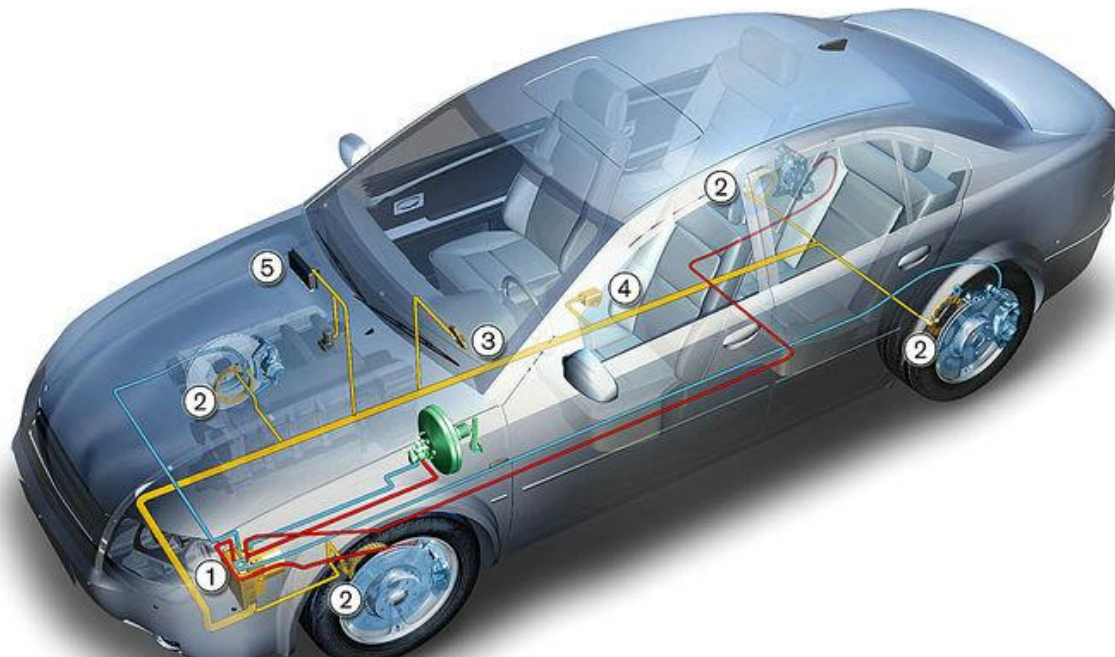


1. Lịch sử phát triển

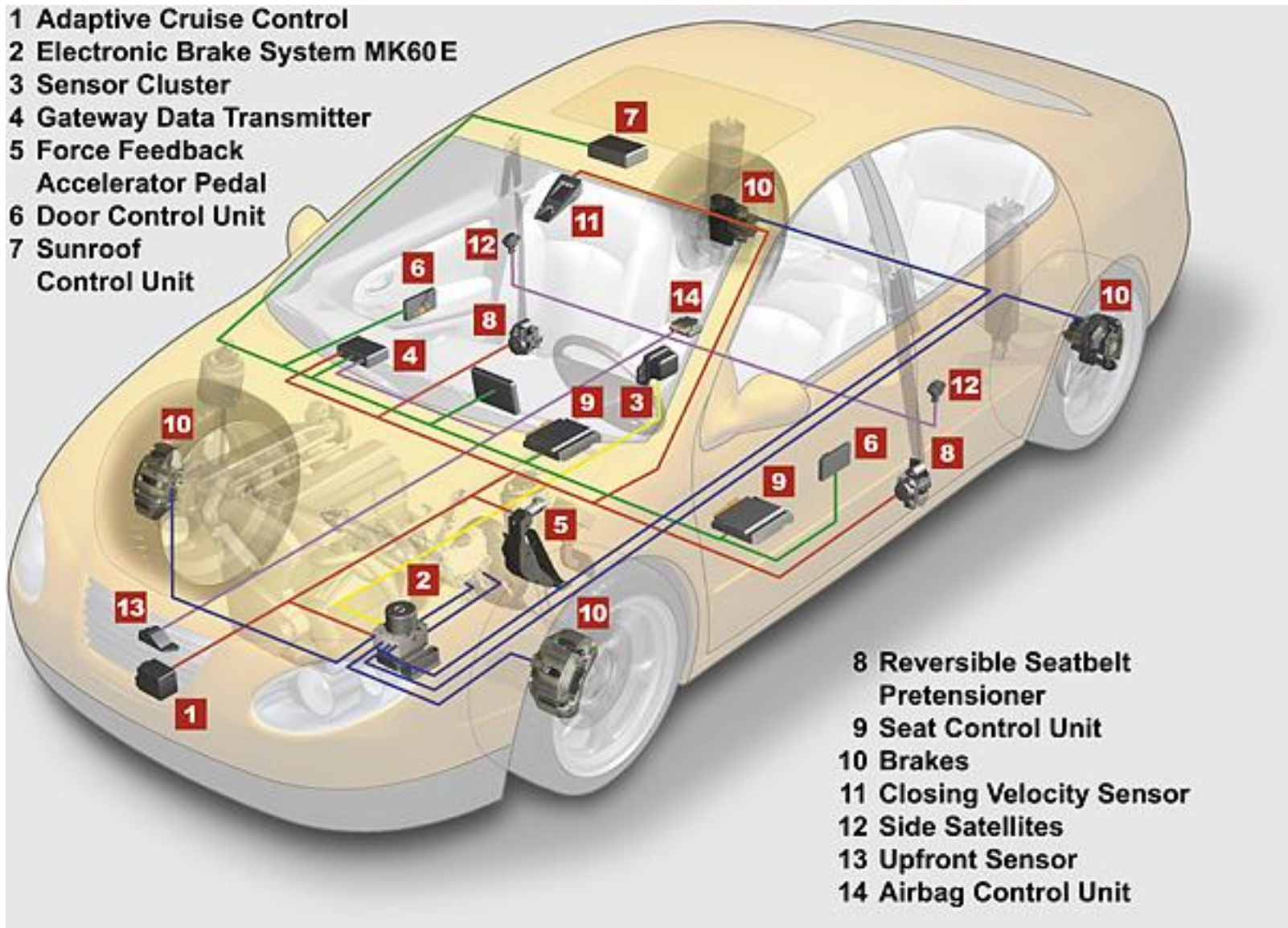
- The BMW 850 coupe was the first CAN Bus vehicle to enter the market in 1986. By reducing the vehicles wiring by 2km, the vehicles overall weight was significantly reduced by at least 50kg and using only half the connectors.
- For the first time, each of the vehicles systems and sensors were able to communicate at very high speeds (25kbps - 1Mbps) on a single or dual-wire communication line as opposed to the previous multi-wire looms

1. Lịch sử phát triển

- In 2006, over 70% of all automobiles sold in North America will utilize CAN Bus technology.
- Beginning in 2008, the Society of Automotive Engineers (SAE) requires 100% of the vehicles sold in the USA to use the CAN Bus communication protocol while the European Union has similar laws



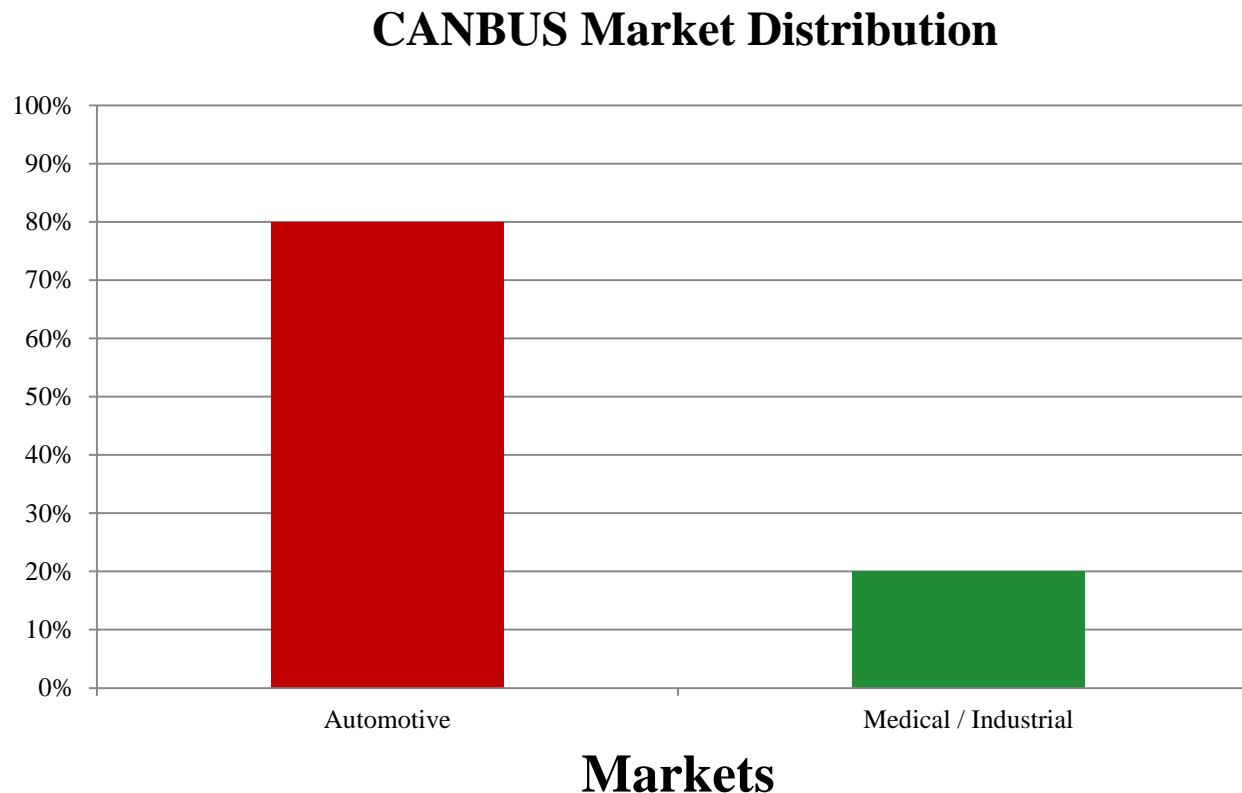
- 1 Adaptive Cruise Control
- 2 Electronic Brake System MK60E
- 3 Sensor Cluster
- 4 Gateway Data Transmitter
- 5 Force Feedback Accelerator Pedal
- 6 Door Control Unit
- 7 Sunroof Control Unit



- 8 Reversible Seatbelt Pretensioner
- 9 Seat Control Unit
- 10 Brakes
- 11 Closing Velocity Sensor
- 12 Side Satellites
- 13 Upfront Sensor
- 14 Airbag Control Unit

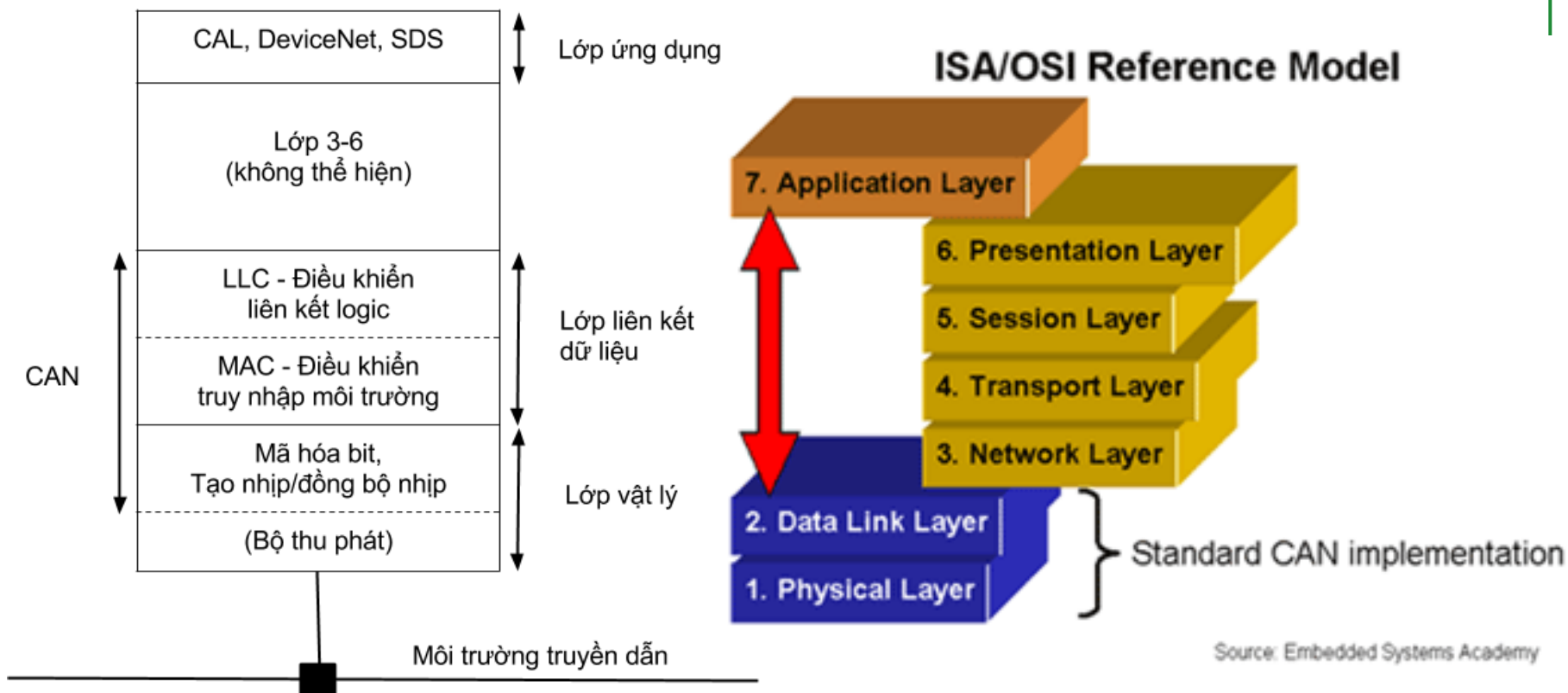
1. Lịch sử phát triển

- Designed specifically for automotive applications
- Today - industrial automation /medical equipment



2. Kiến trúc giao thức

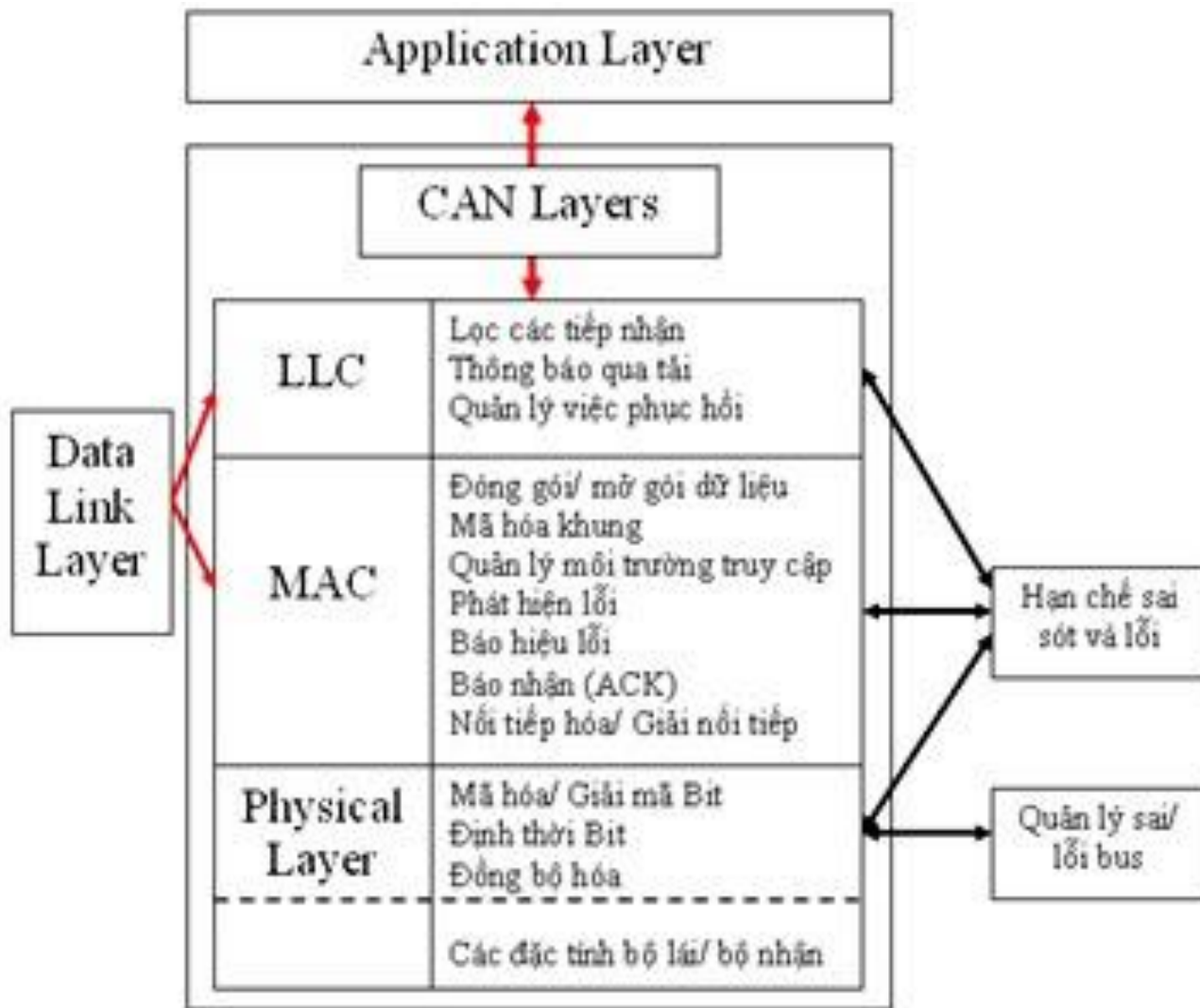
- CAN định nghĩa lớp liên kết dữ liệu (LLC và MAC) và phần chính của lớp vật lý.



- Ngoài ra còn phát triển giao thức lớp cao cho CAN (CANopen) thuộc lớp ứng dụng (Application Layer)

2 Kiến trúc giao thức

■ Giao thức CAN



2 Kiến trúc giao thức

■ Lớp vật lý:

- ❖ Đề cập việc truyền tín hiệu; định nghĩa phương pháp định thời, tạo nhịp bit, mã hóa bit, đồng bộ hóa.
- ❖ Tuy nhiên chuẩn CAN không qui định các đặc tính của các bộ thu phát, với mục đích cho phép lựa chọn môi trường truyền cũng như mức tín hiệu thích hợp cho từng lĩnh vực ứng dụng

■ Lớp liên kết dữ liệu

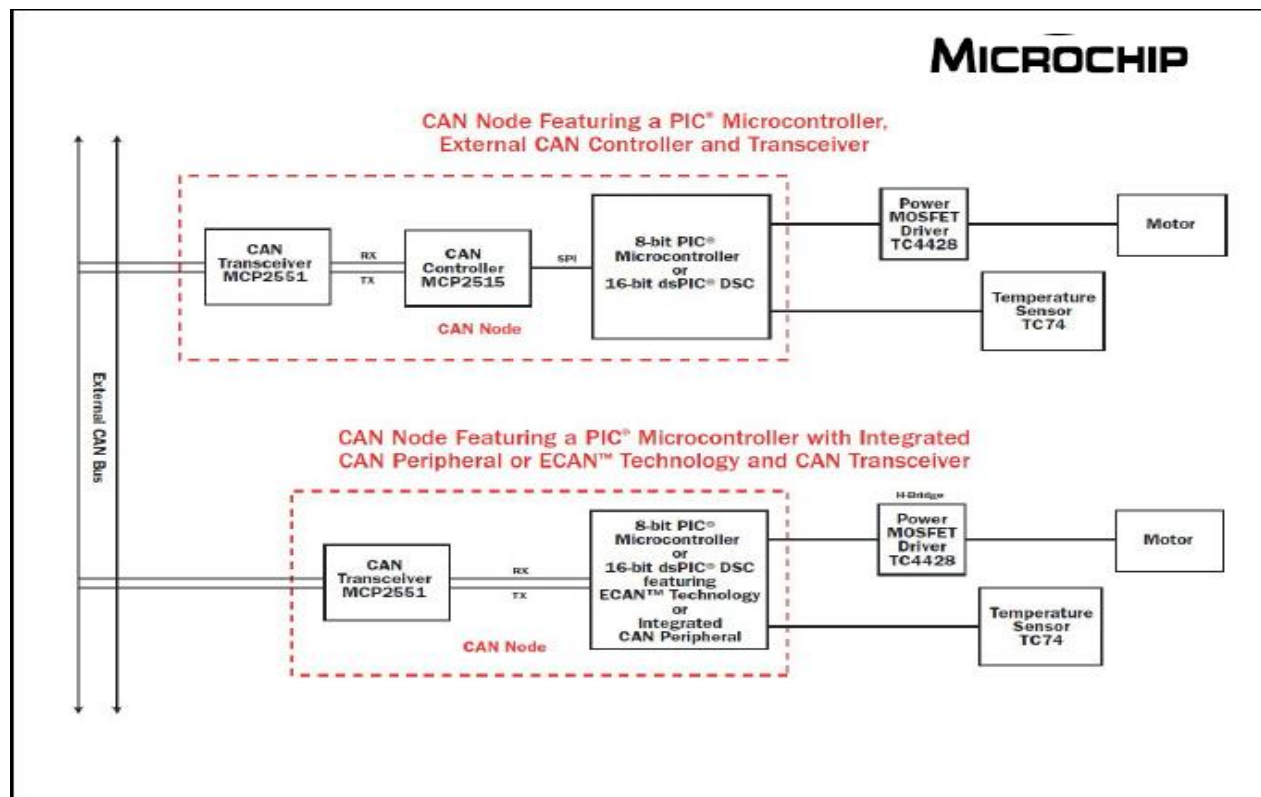
- ❖ Lớp điều khiển truy nhập môi trường (MAC): tạo khung thông báo, điều khiển truy nhập môi trường, xác nhận thông báo và kiểm soát lỗi.
- ❖ Lớp điều khiển liên kết logic (LLC): dịch vụ gửi dữ liệu và yêu cầu dữ liệu từ xa, thanh lọc thông báo, báo cáo tình trạng quá tải và phục hồi trạng thái.

2 Kiến trúc giao thức

- Trong phiên bản CAN2.0B, đặc tả CAN chỉ định nghĩa lớp MAC và một phần lớp LLC. Trong các phiên bản trước đó, hai lớp con của lớp liên kết dữ liệu còn được gọi là lớp đối tượng (Object Layer) và lớp truyền (Transfer Layer).
- Trong các hệ thống bus tiêu biểu xây dựng trên cơ sở CAN như CANOpen, DeviceNet và SDS, giao thức và các dịch vụ của lớp ứng dụng được định nghĩa cụ thể.

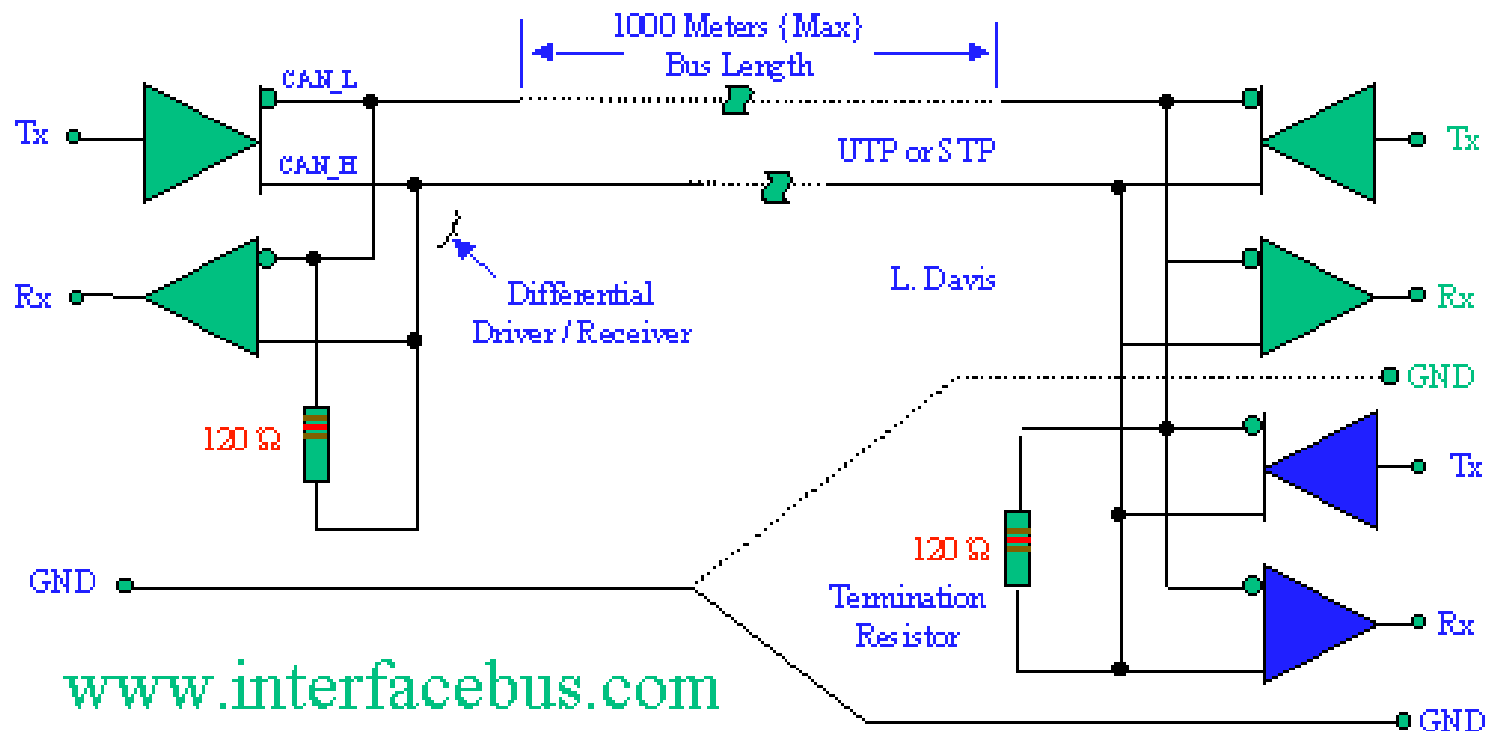
3. Cấu trúc mạng và kỹ thuật truyền dẫn

- CAN thực chất là chuẩn giao thức từ phần trên của lớp vật lý cho đến hết lớp liên kết dữ liệu, vì vậy không qui định cụ thể về chuẩn truyền dẫn cũng như môi trường truyền thông.



3. Cấu trúc mạng và kỹ thuật truyền dẫn

- The CAN bus [CANbus] truyền dẫn cân bằng sử dụng cáp đôi dây xoắn loại Shielded Twisted Pair (STP), Unshielded Twisted Pair (UTP), or Ribbon cable



CAN Bus Electrical Interface Circuit

3. Cấu trúc mạng và kỹ thuật truyền dẫn

- Cấu trúc đường thẳng mắc theo kiểu đường trục/đường nhánh (trunkline/ dropline) với chiều dài đường nhánh $< 0.3m$

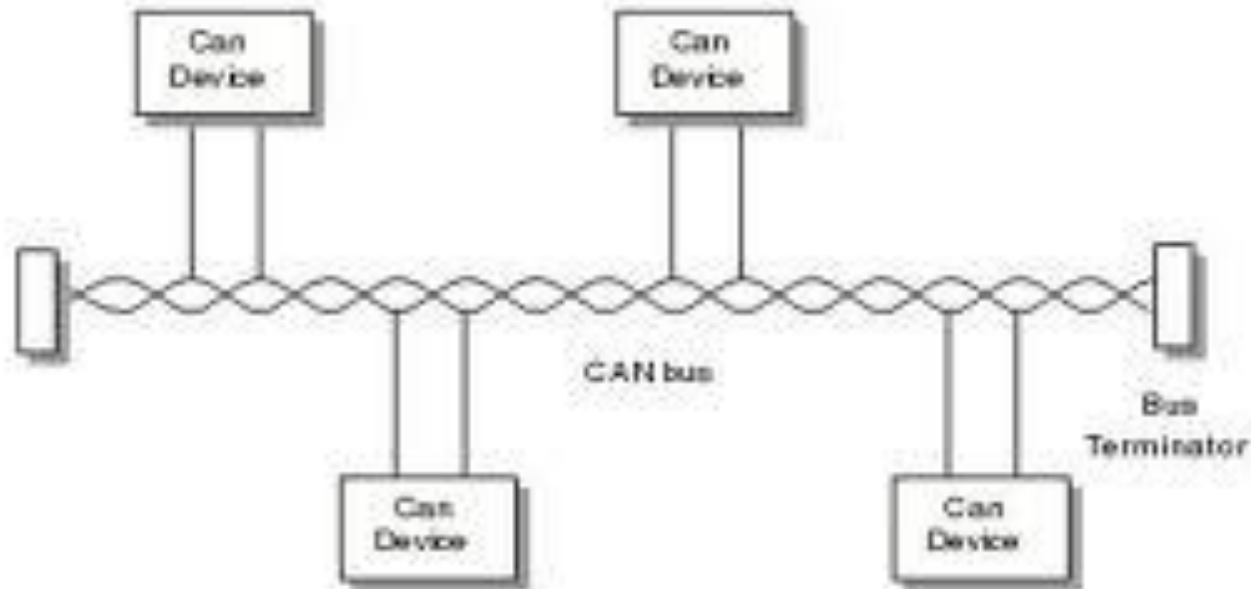
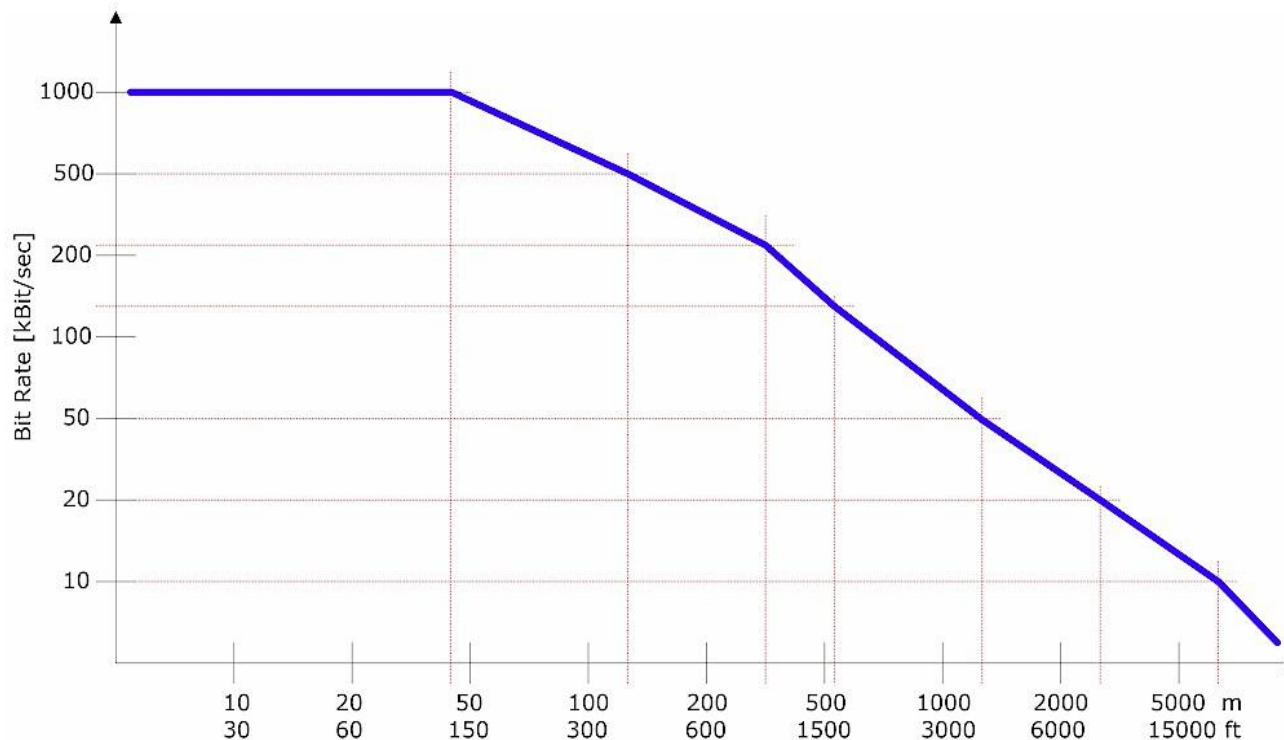


Figure 12-1: A CAN bus

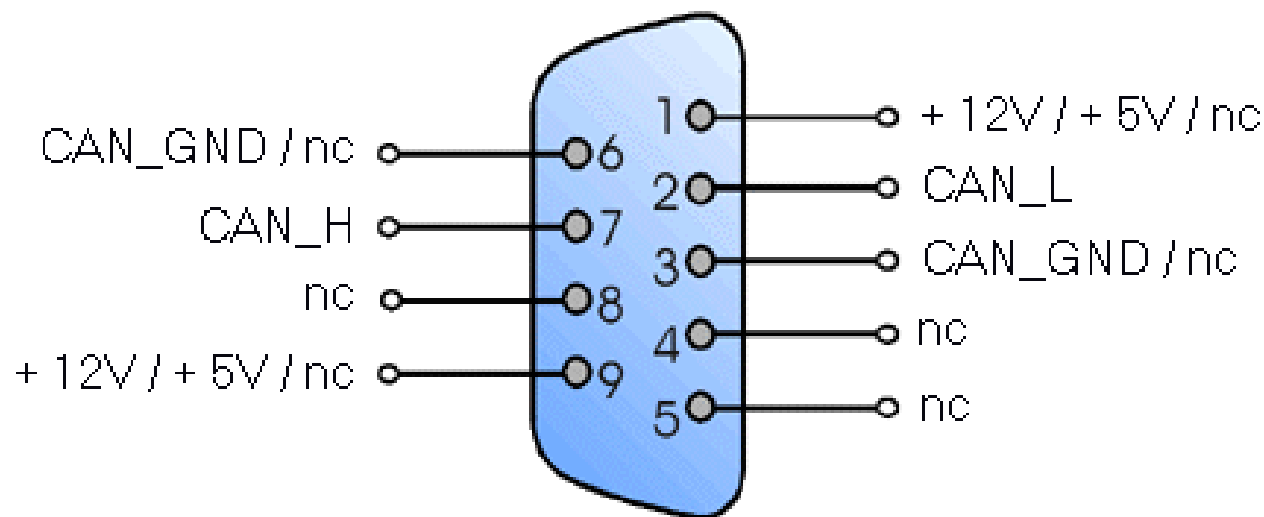
Transmission Characteristics

- ▶ Common baud rates: 1 MHz, 500 KHz and 125 KHz
- ▶ All nodes – same baud rate
- ▶ Max length: 120' to 15000' (rate dependent)
- ▶ Tốc độ truyền: tối đa 1Mbit/s ở khoảng cách 40m và 50kbit/s ở khoảng cách 1000m.



3. Cấu trúc mạng và kỹ thuật truyền dẫn

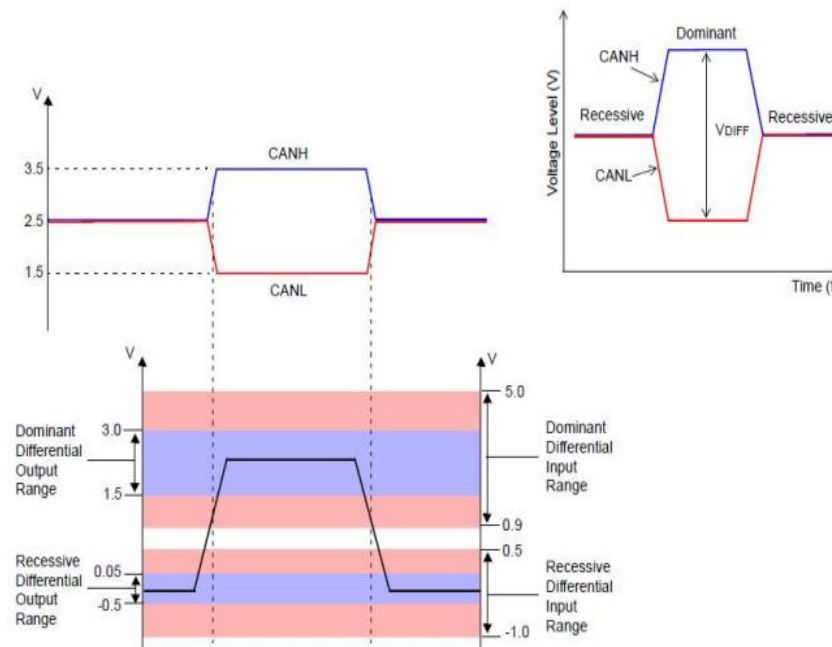
- Số trạm phụ thuộc vào cấu trúc mạng, cáp truyền và đặc tính điện học của các bộ thu phát, với cấu trúc đường thẳng, cáp đôi dây xoắn là hạn chế ở 64 trạm.
- Đường dây bus kết thúc bằng điện trở 120 ohm (thấp nhất là 108 ohm và tối đa là 132 ohm) ở mỗi đầu dây
- Jack kết nối DE-9



3. Cấu trúc mạng và kỹ thuật truyền dẫn

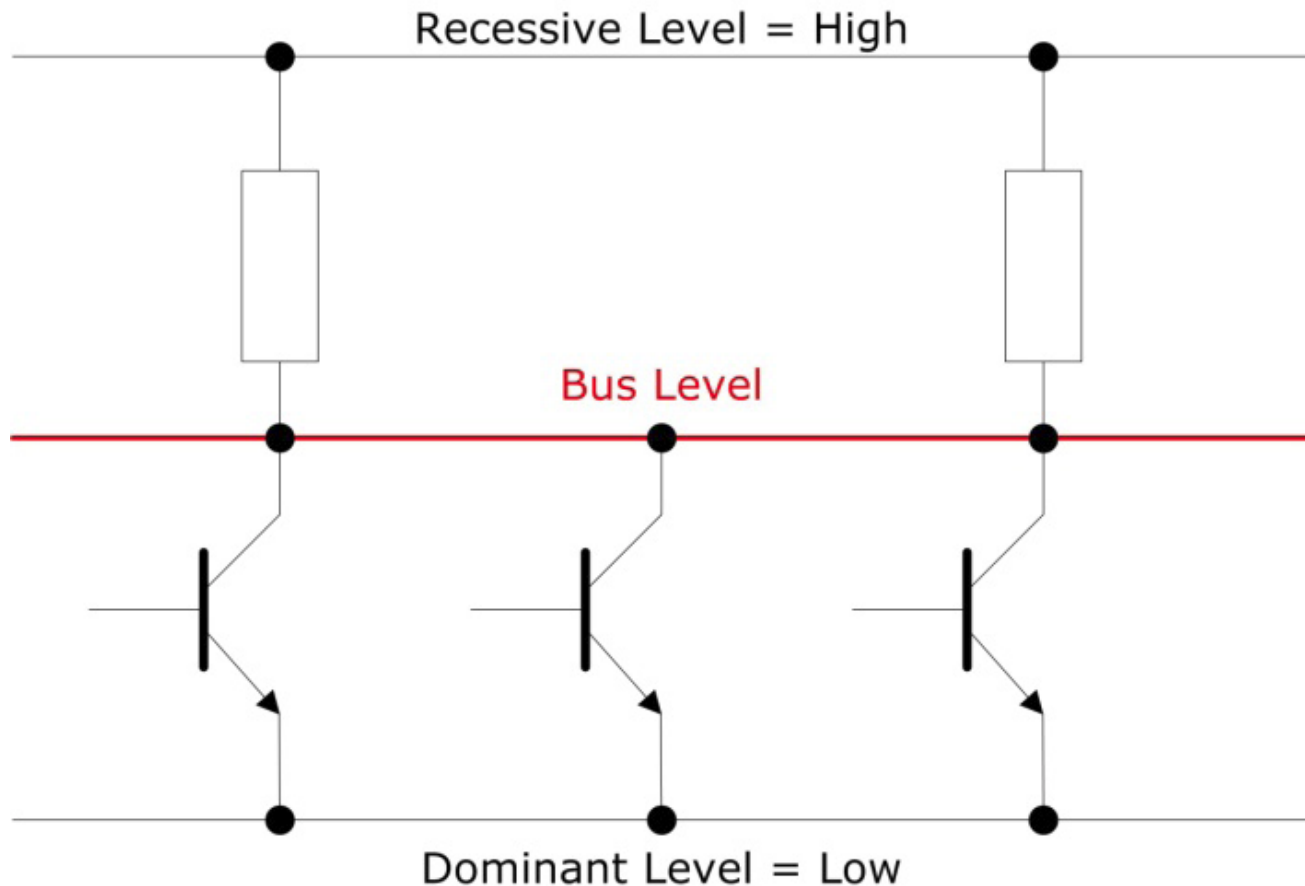
- CAN phân biệt 2 trạng thái logic của tín hiệu là mức trội (dominate) và mức lặn (recessive)
- Trạng thái vật lý thể hiện mức logic không qui định rõ ràng giá trị bit nào ứng với mức tín hiệu nào

ISO11898 NOMINAL BUS LEVELS



Bus arbitration

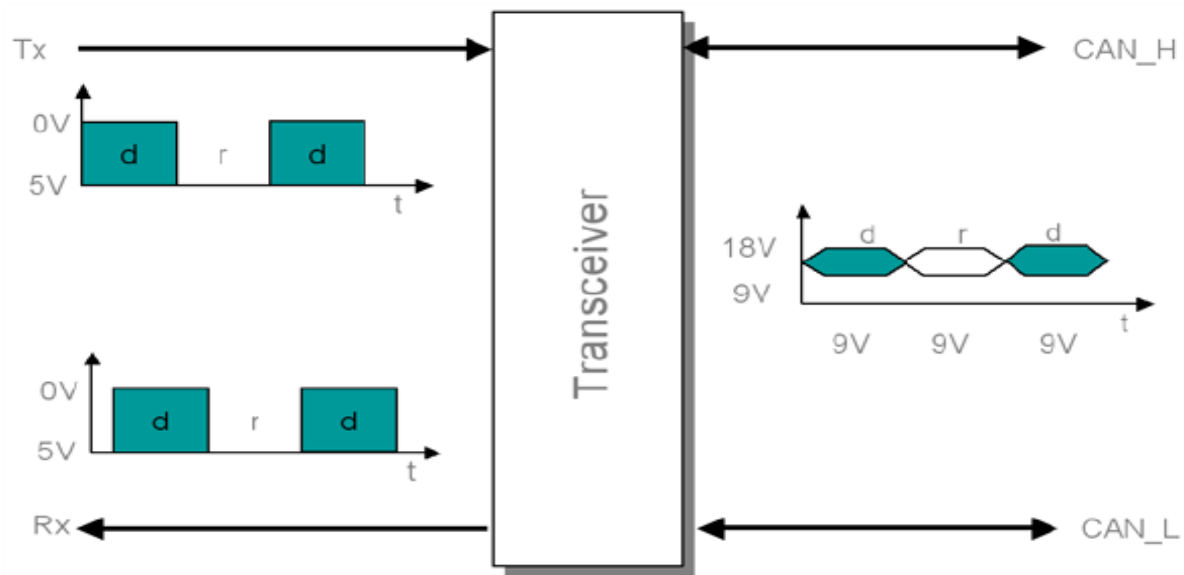
A “0” (low voltage) on the bus by 1 node wins over a “1” (high voltage) on the bus.



3. Cấu trúc mạng và kỹ thuật truyền dẫn

- Trong trường hợp bit trội và bit lặn được phát đồng thời thì bị trội sẽ lấn át và tín hiệu trên bus sẽ có mức trội
- Trong thực tế, nếu sử dụng mạch AND thì mức trội tương ứng với mức 0 và mức lặn tương ứng với mức 1

CAN bus Nominal ISO 11992 Bus Levels



3. Cấu trúc mạng và kỹ thuật truyền dẫn

CAN bus Cables and Connectors

Bus speed	Cable type	Cable resistance/m	Terminator	Bus Length
50 kbit/s at 1000 m	0.75 ... 0.8 mm ² AWG18	70 mOhm	150 ... 300 Ohm	600 ... 1000 m
100 kbit/s at 500 m	0.5 ... 0.6 mm ² AWG20	< 60 mOhm	150 ... 300 Ohm	300 ... 600 m
500 kbit/s at 100 m	0.34 ... 0.6 mm ² AWG22, AWG20	< 40 mOhm	127 Ohm	40 ... 300 m
1000 kbit/s at 40 m	0.25 ... 0.34 mm ² AWG23, AWG22	< 26 mOhm	124 Ohm	0 ... 40 m

Pin-Assignment:



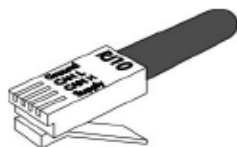
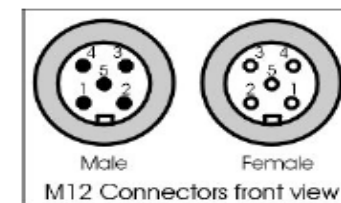
Signals:

Pin	Signal
5	(CAN_V+)
4	CAN_H
3	(CAN_SHLD)
2	CAN_L
1	CAN_GND

Pin	Description	Colour
1	V-	Black
2	CAN_L	Red
3	CAN_H	Green
4	V+	Yellow

Pin	Description	Colour
1	nc	-
2	V-	Black
3	CAN_L	Red
4	CAN_H	Green
5	V+	Yellow
6	nc	-

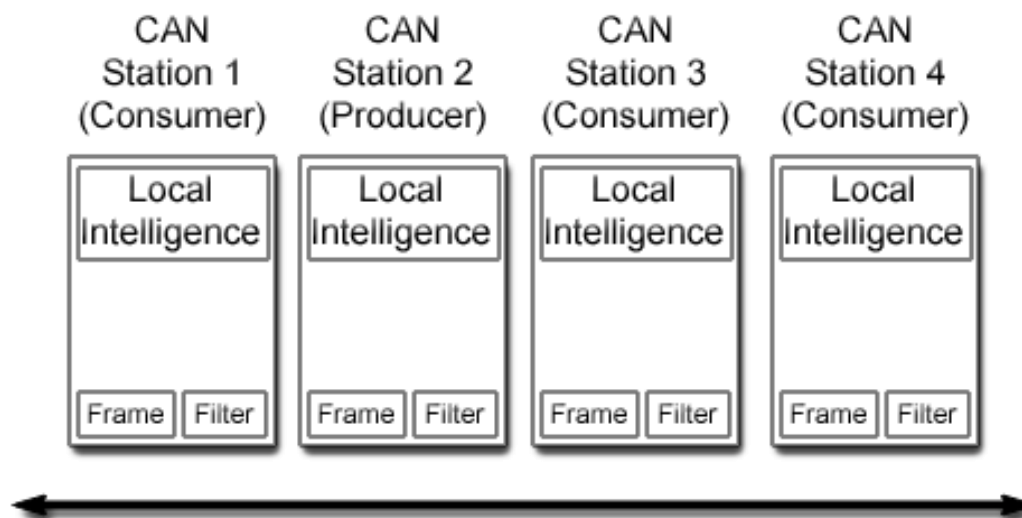
Pin	Description	Color
1	Drain	Bare
2	V+	Red
3	V-	Black
4	CAN_H	White
5	CAN_L	Blue



Tốc độ truyền có thể lựa chọn nhiều mức khác nhau, tuy nhiên phải thống nhất và cố định trong toàn bộ mạng.

4. Cơ chế giao tiếp

- Phương pháp định địa chỉ và giao tiếp hướng đối tượng.
- Đối tượng là thông tin trao đổi trong mạng được được gán một mã căn cước.
- Thông tin được gửi lên bus theo kiểu truyền thông báo. Thông qua phương thức lọc thông báo, trạm nào có nhu cầu thì sẽ tiếp nhận thông báo dựa trên mã căn cước.



4. Cơ chế giao tiếp

- Một trạm có thể yêu cầu trạm khác gửi dữ liệu bằng cách gửi khung REMOTE FRAME. Trạm có khả năng cung cấp nội dung thông tin đó sẽ gửi trả lại một khung dữ liệu DATA FRAME có cùng mã căn cước với khung yêu cầu.
- Giữa DATA FRAME và REMOTE FRAME có cùng ID thì DATA FRAME sẽ được ưu tiên.
- Truy nhập BUS: CSMA/CA (điều khiển phân kênh theo từng bit). Mức ưu tiên dựa vào tính cấp thiết của thông báo và được đặt cố định ID
- Thông báo có mã căn cước càng nhỏ thì ưu tiên càng cao.

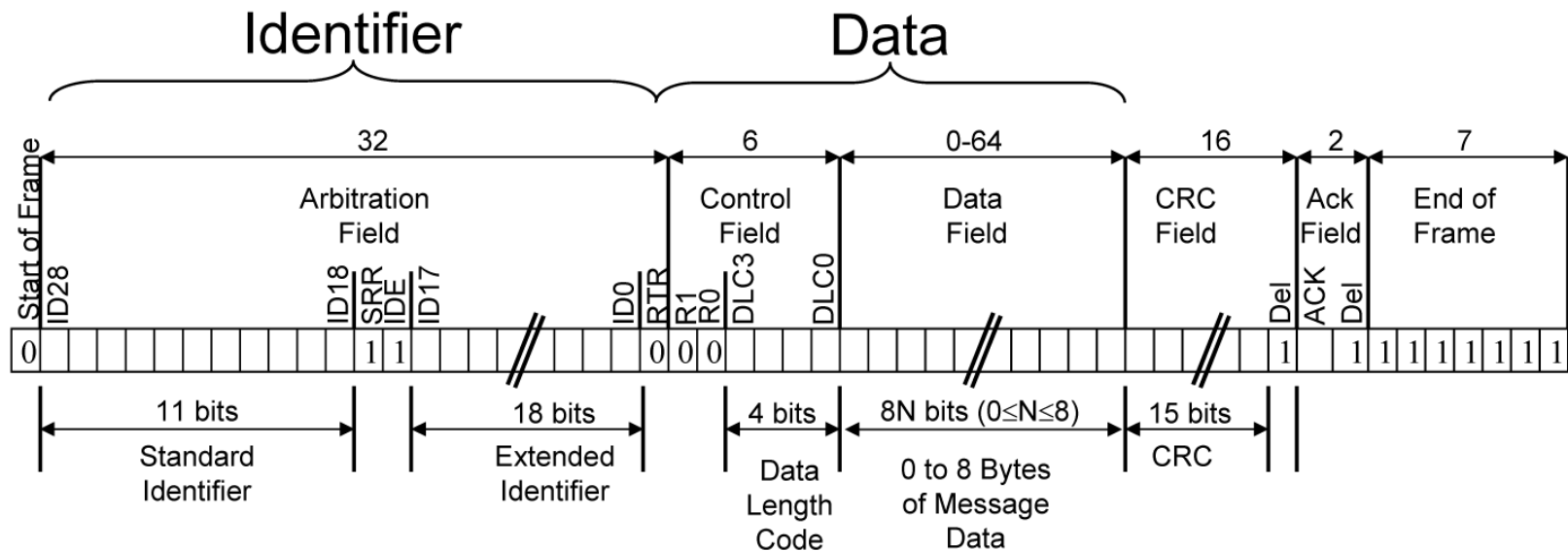
4. Cơ chế giao tiếp

- Bất cứ một trạm nào cũng có thể gửi thông báo mỗi khi đường truyền rỗi.
- Mỗi bức điện bắt đầu bằng bit khởi điểm và mã căn cước.
- Việc xung đột do nhiều trạm cùng gửi sẽ được dựa theo bit của ID.
- Đơn giản, linh hoạt: khi bổ sung hay bỏ 1 trạm thì không cần thay đổi cấu hình ở các trạm khác.
- Một trạm CAN không cần biết thông tin cấu hình hệ thống (VD địa chỉ trạm).
- Tính nhất quán dữ liệu của hệ thống được đảm bảo qua các phương pháp gửi đồng loạt và xử lý lỗi.

5. Cấu trúc bức điện

- CAN định nghĩa 4 kiểu bức điện sau:
 - ❖ Khung dữ liệu (KDL) (**Data Frame**): Mang dữ liệu từ một trạm truyền tới các trạm nhận
 - ❖ Khung yêu cầu dữ liệu (KYCDL) (**Remote Frame**): Gửi từ 1 trạm yêu cầu truyền khung dữ liệu với cùng mã căn cước.
 - ❖ Khung lỗi (**Error Frame**): Gửi từ một trạm bất kì phát hiện lỗi.
 - ❖ Khung quá tải (**Overload Frame**): Tạo khoảng cách thời gian bổ sung giữa hai khung dữ liệu hoặc khung yêu cầu dữ liệu trong trường hợp một trạm bị quá tải.
- Giữa hai KDL or KYCDL cần khoảng cách ít nhất 3 bit lặn để phân biệt

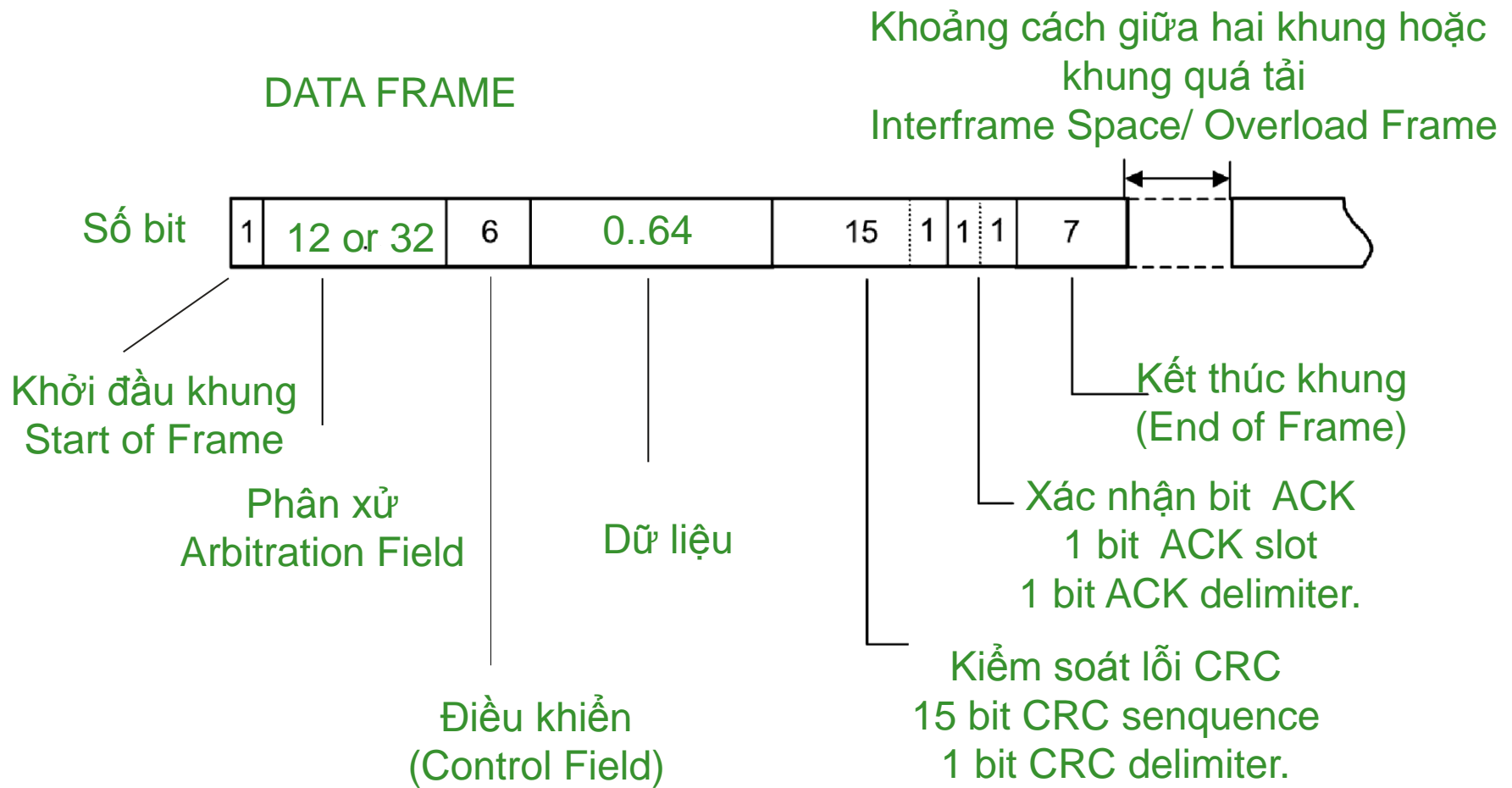
5. Cấu trúc bức điện



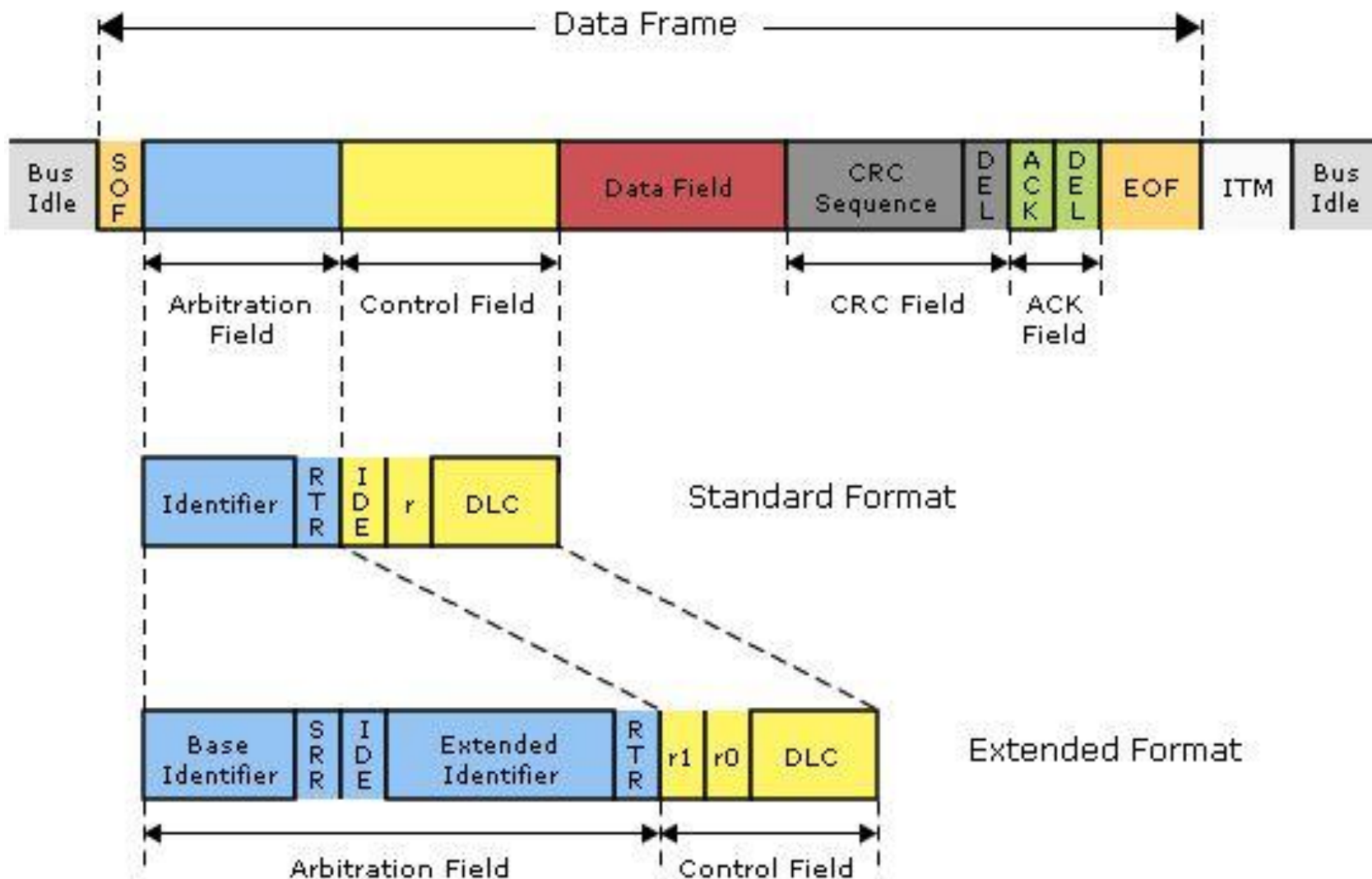
- SOF – Start of Frame
- Identifier – Tells the content of message and priority
- RTR – Remote Transmission Request
- IDE – Identifier extension (distinguishes between CAN standard, 11 bit identifier, and CAN extended, 29 bit identifier.)
- DLC – Data Length Code
- Data – holds up to 8 bytes of data
- CRC – “Cyclic Redundant Check” sum
- ACK – Acknowledge
- EOF – End of Frame
- IFS – Intermission Frame Space. Minimum number of bits separating consecutive messages.

5. Cấu trúc bức điện

A. Khung dữ liệu/yêu cầu dữ liệu (DATA/REMOTE FRAME):



5. Cấu trúc bức điện



5. Cấu trúc bức điện

- Khởi đầu khung là một bit trội. Tất cả các trạm sẽ phải đồng bộ hóa bit khởi đầu này.
- Ô phân xử được sử dụng là mức ưu tiên của bức điện, chiều dài 12 bit (Standard Frame) 32 bit (Extended frame), trong đó mã căn cước dài 11 hoặc 29 bit. Bit cuối gọi là bit RTR (*Remote Transmission Request*), với KDL thì là bit trội, khung yêu cầu dữ liệu là bit lặn

Khung chuẩn (Standard Frame)	12 bit		
	11 bit mã căn cước		1 bit RTR
Khung mở rộng (Extended frame)	32 bit		
	29 bit mã căn cước	2 bit	1 bit RTR

5. Cấu trúc bức điện

- Ô điều khiển dài 6 bit, 4 bit cuối mã hóa chiều dài dữ liệu (bit trội 0, bit lặn 1).

6 bit	
2 bit tùy thuộc khung chuẩn hay mở rộng	4 bit mã hóa chiều dài dữ liệu

- ❖ Standard Frame: Bit đầu tiên là bit IDE được truyền dạng dominant (trội), bit thứ hai là bit dành riêng r_0 thì tùy ý
- ❖ Extended Frame: Hai bit đầu tiên là hai bit dành riêng r_1 và r_0 . Hai bit này thường gửi dạng dominant (trội) nhưng các bộ nhận cho phép hai bit này có thể tùy ý là dominant (trội) hoặc recessive (lặn).

5. Cấu trúc bức điện

- Ô dữ liệu: có chiều dài 0...8 byte, mỗi byte truyền đi theo thứ tự MSB đến bit LSB

0..8 byte						
Byte 0			...	Byte k		
MSB0	...	LSB0	...	MSBk	...	LSBk

- Ô kiểm soát lỗi CRC:
 - gồm 15 bit chứa CRC và 1 bit lặn phân cách;
 - dãy bit đầu vào để tính CRC gồm bit khởi đầu khung, ô phân xử, ô điều khiển và ô dữ liệu;
 - đa thức phát

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

5. Cấu trúc bước điện

- Ô xác nhận ACK (Acknowledgegment) gồm 2 bit, được phát đi là các bit lặn. Mỗi trạm nhận được bước điện phải kiểm tra mã CRC, nếu đúng sẽ phát chông 1 bit trội trong thời gian nhận được bit ACK Slot. Vậy sẽ có 1 bit trội giữa 2 bit lặn phân cách.
- Kết thúc khung được đánh dấu bằng 7 bit lặn.

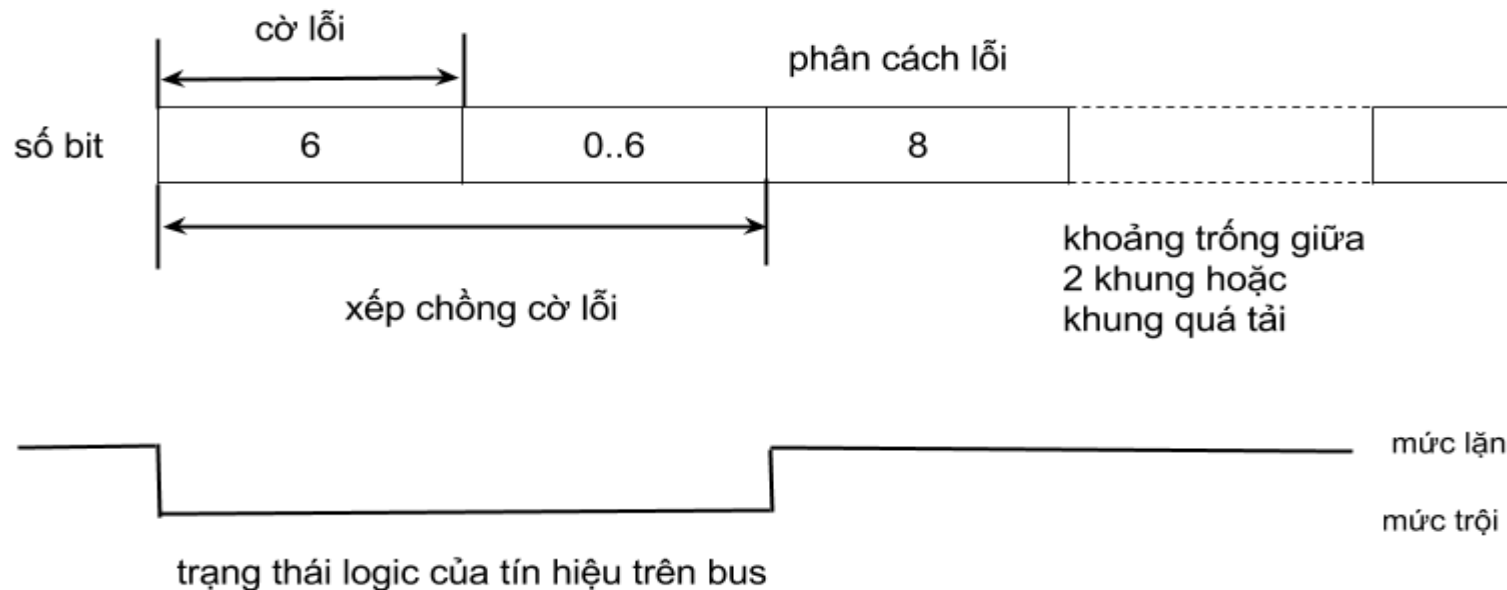


Remote Frame có cấu trúc tương tự Data Frame, nhưng không mang dữ liệu và khác nhau ở bit cuối của ô phân xử.

5. Cấu trúc bức điện

➤ Khung lỗi

- Được gửi từ bất kỳ trạm nào phát hiện ra lỗi.
- Gồm cờ lỗi và phân cách lỗi.



5. Cấu trúc bức điện

➤ Cờ lỗi

- Loại lỗi chủ động – cờ lỗi chủ động gồm 6 bit trội liên nhau,
 - ❖ Một trạm ‘error active’ khi phát hiện điều kiện lỗi sẽ báo hiệu bằng cách gửi một cờ lỗi chủ động.
 - ❖ Dạng của cờ lỗi vi phạm luật chèn bit (Bit Stuffing) được dùng cho tất cả các vùng từ SOF đến CRC delimiter hay phá hủy định dạng cố định của vùng ACK hay EOF. Kết quả là tất cả các trạm khác phát hiện một điều kiện lỗi và chúng sẽ bắt đầu truyền một cờ lỗi
- Loại lỗi bị động – cờ lỗi bị động gồm 6 bit lặn liên nhau.
 - ❖ Xếp chồng cờ lỗi do các trạm phát hiện lỗi sau và cũng gửi khung lỗi.

5. Cấu trúc bức điện

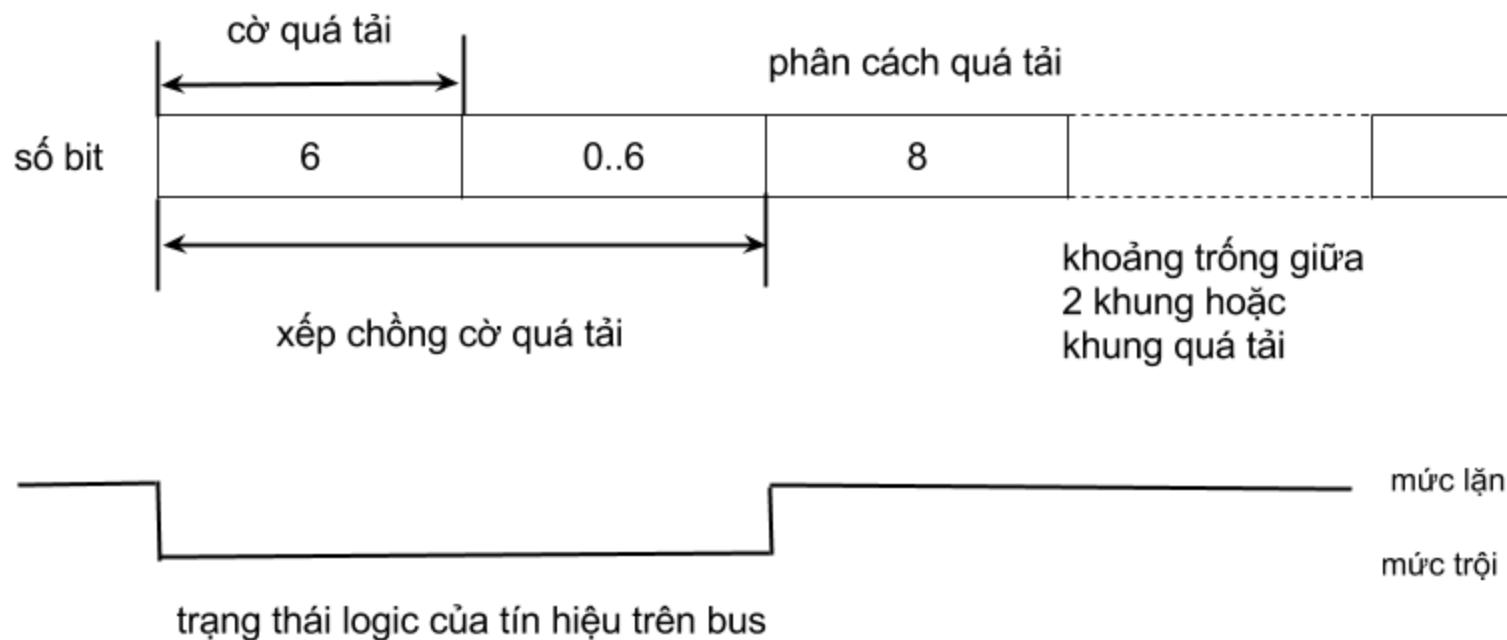
➤ Cờ lỗi

■ Phân cách lỗi gồm 8 bit lặn liên nhau

- ❖ Sau khi gửi xong 1 cờ lỗi, mỗi trạm gửi tiếp một số bit lặn, đồng thời quan sát bus. Đến khi phát hiện bit lặn (trạm khác gửi xong cờ lỗi chủ động), chúng gửi tiếp 7 bit còn lại.

5. Cấu trúc bức điện

- Khung quá tải
 - Gồm cờ quá tải và phân cách quá tải



5. Cấu trúc bức điện

➤ OVERLOAD FRAME:

■ Gồm Overload Flag và Overload Delimiter

❖ Cờ quá tải (Overload Flag) gồm 6 bit trội liên tục. nhau, phá bỏ dạng cố định của ô INTERMISSION ở khoảng trống giữa hai khung, dẫn đến tất cả các trạm khác phát hiện tình trạng quá tải và bắt đầu gửi cờ quá tải.

❖ Phân cách quá tải (Overload Delimiter) được đánh dấu bằng 8 bit lặn liên tục. Sau khi gửi xong một cờ, mỗi trạm phải quan sát cho đến khi phát hiện ra một bit lặn, khi đó chúng sẽ phát tiếp bảy bit lặn.

■ Xếp chồng cờ quá tải do các trạm phát hiện quá tải sau và cũng gửi khung quá tải.

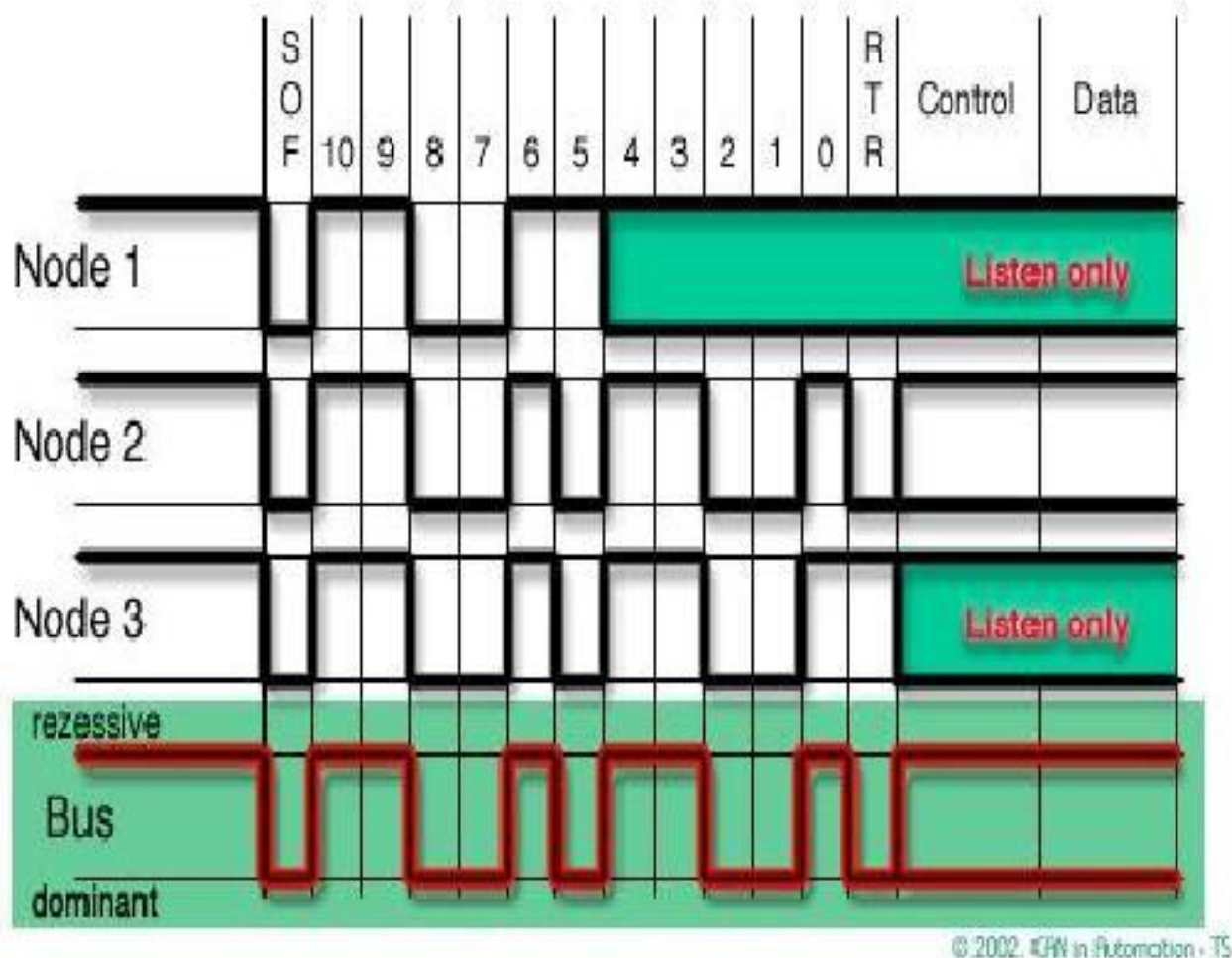
6.Truy nhập BUS

A. Phương pháp

- ❖ CAN sử dụng phương pháp môi trường CSMA/CA, tức là điều khiển phân kênh theo từng bít.
- ❖ Phương pháp phân mức ưu tiên truy nhập bus dựa theo tính cấp thiết của nội dung thông báo. Mức ưu tiên này phải được đặt cố định trước khi hệ thống đưa vào vận hành.
- ❖ Mã căn cước không những mang ý nghĩa của dữ liệu mà còn đồng thời được sử dụng là mức ưu tiên.
- ❖ Bất cứ một trạm nào trong mạng cũng có thể bắt đầu gửi thông báo khi bus rỗi.

6. Truy nhập BUS

- Ví dụ hệ thống nhiều node tham gia vào truy cập bus



6. Truy nhập BUS

■ Giải quyết khi có xung đột

- ❖ Mỗi bước điện đều bắt đầu bằng bit start và mã căn cước nên việc xung đột xảy ra sẽ được phân xử dựa trên mã căn cước. Bộ thu phát sẽ có nhiệm vụ so sánh mức tín hiệu gửi đi và quan sát trên đường truyền, để phân xử.
- ❖ Trong trường hợp có xảy ra va chạm giữa thông báo mang dữ liệu (DATA FRAME) và thông báo yêu cầu gửi dữ liệu (REMOTE FRAME) thì thông báo mang dữ liệu sẽ được ưu tiên.

7. Mã hóa dữ liệu

- ❖ CAN được sử dụng phương pháp nhồi bit (bit stuffing) trước khi được chuyển đổi thành tín hiệu trên đường truyền.
 - ❖ Dãy bit đầu vào bao gồm:
 - Bit khởi đầu khung.
 - Ô phân xử.
 - Ô điều khiển.
 - ❖ Dữ liệu và dãy CRC.
- ❖ Không thực hiện với các phần còn lại của KDL, YCDL, KL, QT

7. Mã hóa dữ liệu

NRZ = Non-Return-To_Zero

- Fewer transitions (on average) = less EMI, but requires less oscillator drift

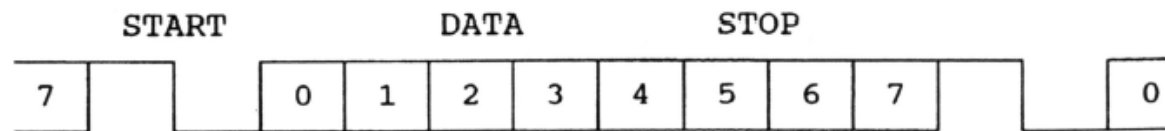
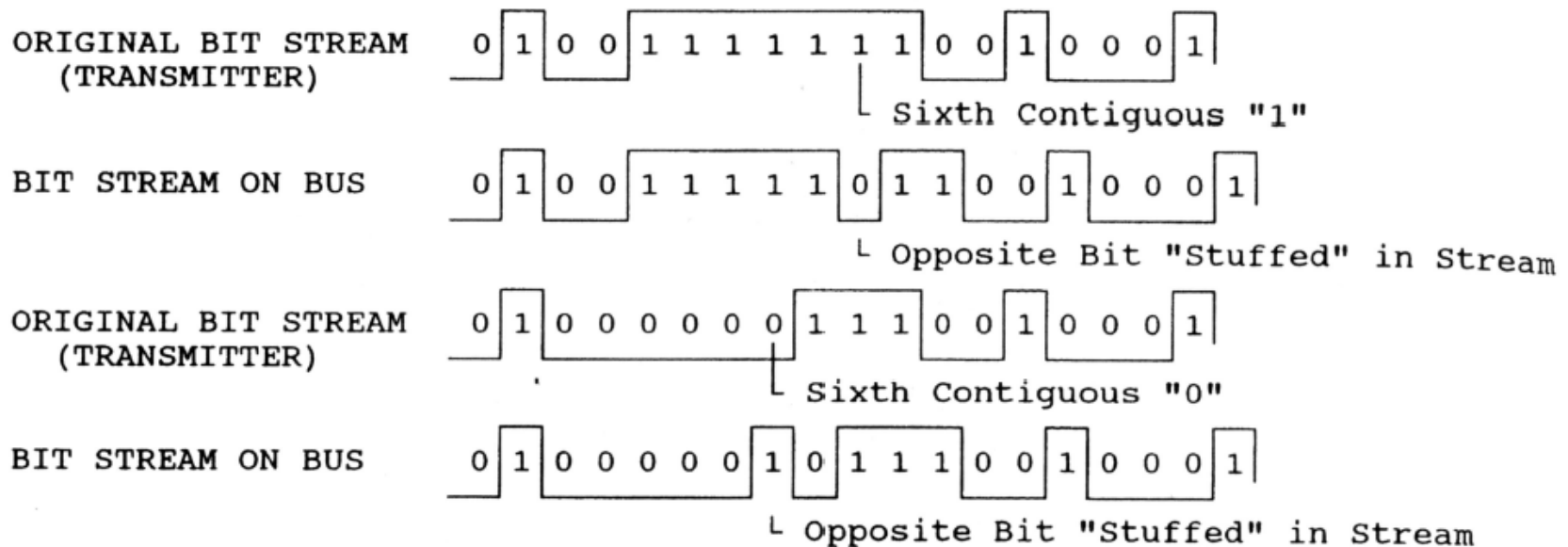


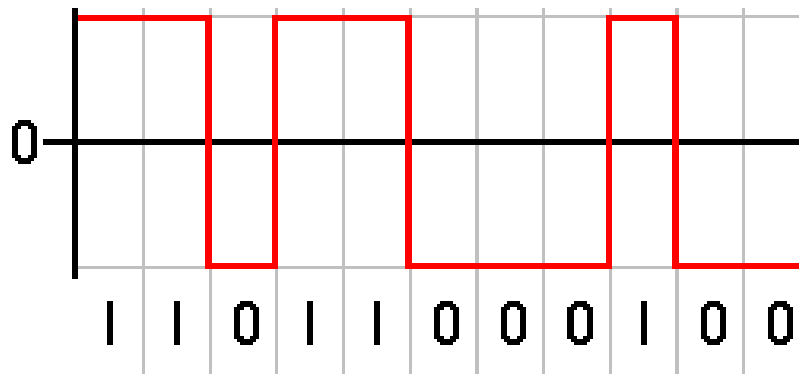
FIGURE 26.21 A 10-bit NRZ waveform (LSB first).

- Bit stuffing relaxes oscillator drift requirements



7. Mã hóa dữ liệu

- ❖ Khi 5 bit liên tục giống nhau bộ phát sẽ bổ sung 1 bit nghịch đảo vào cuối. Bên nhận sẽ phát hiện bit được nhồi và tái tạo thông tin.
- ❖ Cuối cùng, dãy bit được mã hóa theo theo phương pháp **NRZ (*Non-return-to-rezo*)**.



8. Bảo toàn dữ liệu

A. MỤC ĐÍCH

- ❖ Nhằm đảm bảo an toàn trong truyền dẫn dữ liệu. mỗi trạm CAN đều sử dụng nhiều biện pháp để tự kiểm tra, phát hiện và báo hiệu lỗi.
- ❖ Các biện pháp được sử dụng ở trong CAN:
 - Theo dõi mỗi mức tín hiệu được truyền đi và so sánh với tín hiệu nhận được trên bus.
 - Kiểm soát qua mã CRC.
 - Thực hiện nhồi بیت.
 - Kiểm soát khung thông báo

8. Bảo toàn dữ liệu

B. Yêu cầu độ an toàn

- ❖ Các trạm bắt buộc phải kiểm tra thông tin nhận được. Khi phát hiện có lỗi, các trạm đều có trách nhiệm truyền khung lỗi. Các thông báo sẽ bị dừng và tự động phát lại.

C. Khả năng nhận biết lỗi kéo dài

- ❖ Các trạm CAN có khả năng phân biệt lỗi nhất thời và lỗi kéo dài, ví dụ 1 trạm bị sự cố. Lúc đấy các trạm hỏng sẽ tự động được tách ra.

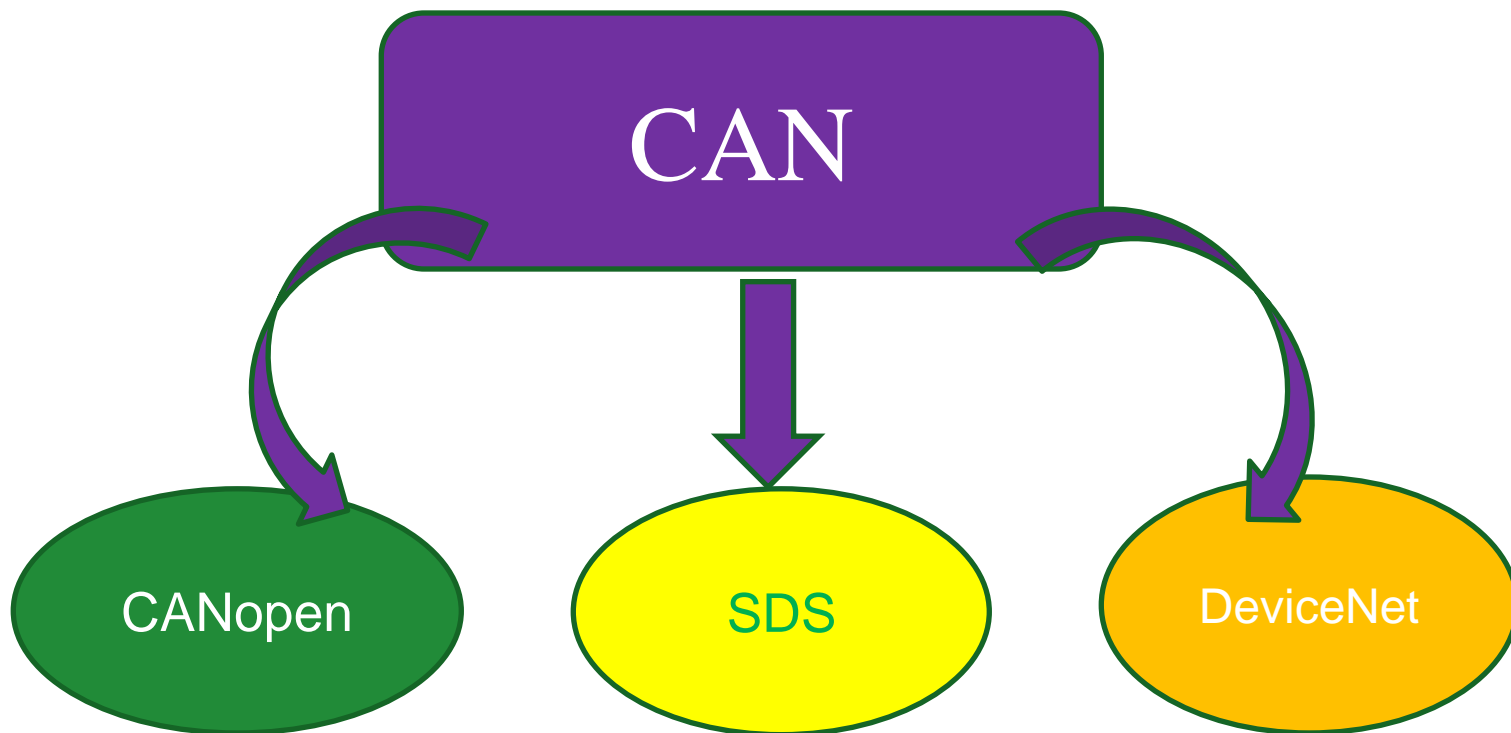
8. Bảo toàn dữ liệu

❖ Hiệu quả đạt được:

- Phát hiện được tất cả các lỗi toàn cục.
- Phát hiện được tất cả các lỗi cục bộ tại bộ phát.
- Phát hiện được 5 bít lỗi phân bố ngẫu nhiên trong 1 bức điện..
- Phát hiện được các lỗi có số bít lỗi là chẵn.
- Các lỗi đột ngột có chiều dài nhỏ hơn 15 bit trong một thông báo
- Tỷ lệ lỗi còn lại $R = 4,7 \cdot 10^{-11}$.

9. Các hệ thống tiêu biểu dựa trên can

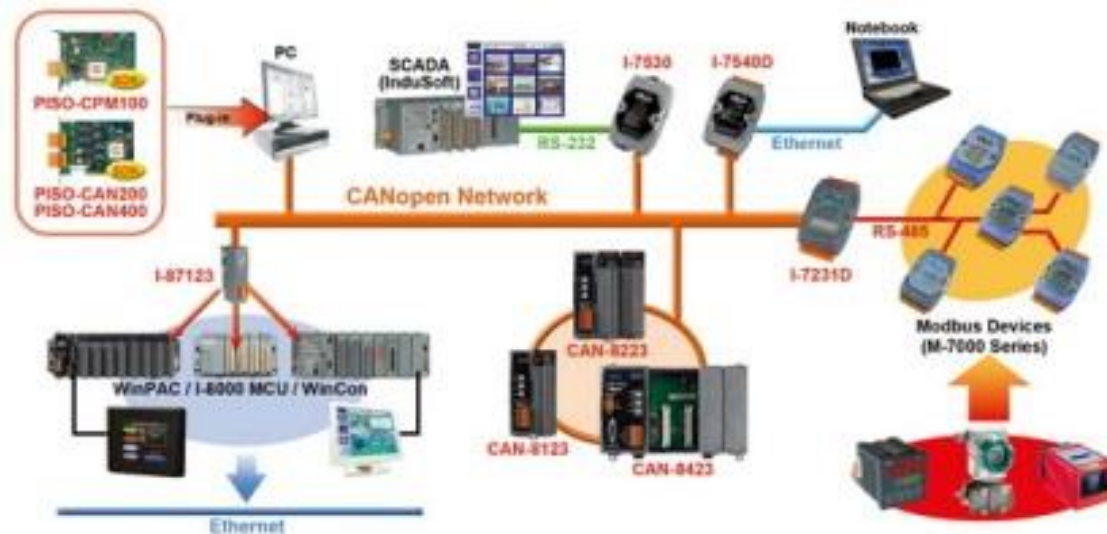
CAN không quy định các giao thức và dịch vụ thuộc lớp ứng dụng, mà để các hệ thống bus thực hiện theo yêu cầu cụ thể khác nhau.



9. Các hệ thống tiêu biểu dựa trên can

A. CANOpen

- CANopen là một hệ thống mạng do tổ chức CAN in Automation phát triển dựa trên CAN, sử dụng lớp vật lý theo chuẩn ISO 11898
- ❖ Các lĩnh vực ứng dụng tiêu biểu là các hệ thống điều khiển chuyển động, các dây chuyền lắp ráp và xử lý nguyên liệu



9. Các hệ thống tiêu biểu dựa trên can

A.CANOpen

- ❖ Ví dụ các thiết bị được nối mạng có thể là các khối cảm biến đa kênh, cảm biến thông minh , van khí nén, bộ đọc mã vạch cơ cấu truyền động và giao diện vận hành.
- ❖ Ưu điểm :
 - Độ tin cậy cao.
 - Hiệu suất sử dụng đường truyền lớn.
 - Hợp với điều khiển chuyển động tốc độ cao.

9. Các hệ thống tiêu biểu dựa trên can

A.CANOpen

❖ Nhược điểm:

- là sự ràng buộc về tốc độ truyền và chiều dài mạng cũng như lượng dữ liệu hạn chế 8 byte trong mỗi bước điện.
- Giao thức của CANOpen khá phức tạp.

9. Các hệ thống tiêu biểu dựa trên can

A.CANOpen

- ❖ CANOpen phát triển 1 họ các qui luật chuẩn giao thức cao cấp cung cấp các chức năng bổ sung như các đối tượng giao tiếp chuẩn cho dữ liệu quá trình, dữ liệu dịch vụ, quản trị mạng,.....
- ❖ CANopen đưa ra các phương pháp hỗ trợ cải thiện tính thời gian thực trong giao tiếp.
- ❖ Đối với các ứng dụng điều khiển chuyển động. CANopen hỗ trợ việc đồng bộ hóa dữ liệu tuần hoàn và không tuần hoàn.

9. Các hệ thống tiêu biểu dựa trên can

A.CANOpen

- ❖ Sử dụng cơ chế giao tiếp hướng đối tượng và cung cấp các chức năng tương tự cho việc truyền dữ liệu thời gian thực, dữ liệu cấu hình và thông tin quản trị mạng. CANopen hướng nhiều theo kiểu thông báo.
- ❖ Ứng dụng:
robot, máy công cụ, máy dệt, máy in, máy đóng bao bì.....

9. Các hệ thống tiêu biểu dựa trên can

B.SDS

- ❖ Đặc điểm tiêu biểu của SDS là tính năng thời gian thực, phù với cả các hệ thống điều khiển có cấu trúc tập trung cũng như phân tán.
- ❖ SDS hoạt động chủ yếu theo hướng sự kiện.
- ❖ Ứng dụng:

Trong các dây chuyền lắp ráp, xử lí nguyên liệu, đóng gói và phân loại sản phẩm. Dựa trên chuẩn truyền dẫn RS-485.

9. Các hệ thống tiêu biểu dựa trên can

B.SDS

❖ Ưu điểm:

- Giá thành thấp.
- Kích cỡ vật lí các mạch điện tử tương đối nhỏ.
- Độ tin cậy cao
- Khả năng chẩn đoán hệ thống và sử dụng hiệu quả đường truyền.

9. Các hệ thống tiêu biểu dựa trên can

B.SDS

❖ Nhược điểm:

Tốc độ truyền cũng như chiều dài đường truyền không lớn lắm, chiều dài dữ liệu trong 1 thông báo hạn chế.

❖ SDS hỗ trợ rất tốt việc chẩn đoán các thiết bị cảm biến thông minh cho phép người vận hành phát hiện các cảm biến cần bảo dưỡng hay sắp hỏng. Các thiết bị có sự cố được thay thế trong khi hệ thống vẫn đang vận hành.

III. Kết luận

- ❖ Tốc độ phát triển nhanh chóng của công nghệ vi điện tử, kỹ thuật truyền thông và công nghệ phần mềm trong những năm gần đây đã tạo sự chuyển hướng cơ bản trong các giải pháp tự động hóa công nghiệp.
- ❖ *Hệ thống mạng CAN (Controller Area Network) là một hệ thống mạng tuyệt vời nhưng lại không mấy được phổ biến tại nước ta cả về mặt tài liệu cũng như mặt ứng dụng. Thực tế thì mạng công nghiệp ở nước ta đều do kỹ sư nước ngoài thiết kế và họ sử dụng ứng dụng CAN rất nhiều.*
- ❖ *Ngày nay, những nhà sản xuất linh kiện bán dẫn như Motorola, Philips, Intel, Atmel, Microchip đã tích hợp CAN vào trong các chip bán dẫn của họ*

6.5 DEVICENET

- DeviceNet phát triển dựa trên CAN, nhưng khác với CAN, mỗi thành viên trong một mạng DeviceNet được đặt một địa chỉ trong khoảng từ 0-63, được gọi là MAC-ID (Medium Access Control Identifier). Việc bổ sung hay bỏ đi một trạm có thể thực hiện ngay khi mạng còn đóng nguồn
- Là đường trục/đường nhánh. Ba tốc độ truyền qui định là 125 Kbit/s, 250 Kbit/s và 500 Kbit/s tương ứng với chiều dài tối đa của đường trục là 500m, 250m và 100m

Giới thiệu mạng DeviceNet

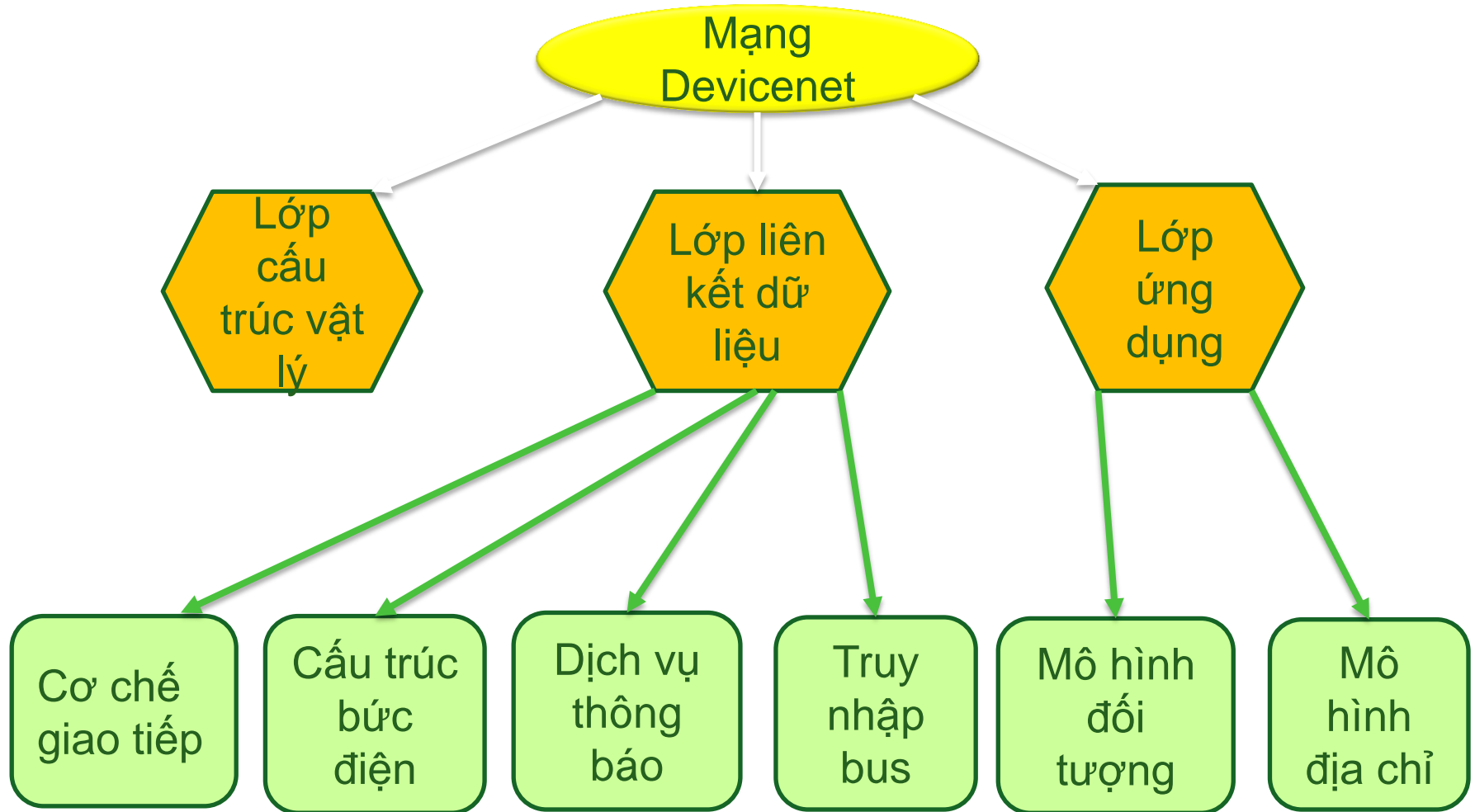
- Được giới thiệu vào năm 1994 bởi hãng Allen-Bradley
- Sau đó được chuyển giao công nghệ cho ODVA (Open DeviceNet Vendor Association) là một tổ chức phi lợi nhuận, được tạo thành từ hàng trăm công ty trên toàn thế giới
- Hiện nay ODVA có trên 300 công ty được đăng kí thành viên và hơn 800 nhà cung cấp dịch vụ

DeviceNet

- Mạng Devicenet dùng để kết nối sensor, biến tần, Remote I/O



DeviceNet



Cấu trúc mạng và kĩ thuật truyền dẫn

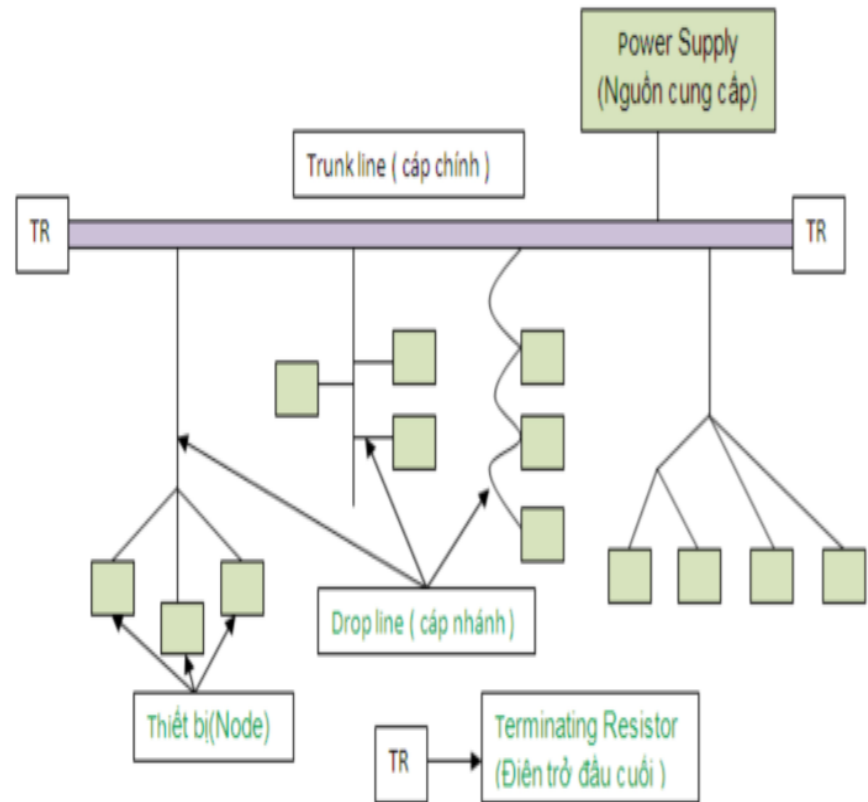
- ✓ DeviceNet có cấu trúc mạng kiểu đường trục/ đường nhánh. Đường trục (các dây màu xanh da trời hoặc trắng) là xương sống (backbone) của mạng và phải được kết thúc với trở đầu cuối là 120 Ohm, 0.25W
- ✓ Các đường nhánh có chiều dài tối đa là 6 m, dùng để kết nối các nút mạng với đường trục chính
- ✓ Trở đầu cuối có tác dụng đánh dấu điểm cuối cùng của mạng
- ✓ Về kĩ thuật truyền dẫn: DeviceNet không quy định cụ thể về chuẩn truyền cũng như môi trường truyền thông
- ✓ Phương pháp mã hóa bit: DeviceNet sử dụng phương pháp mã hóa bit của CAN

Cấu trúc mạng và kĩ thuật truyền dẫn

■ Tốc độ và cấu trúc

Tốc độ	125 kps	250 kps	500 kps
Chiều dài tối đa của cáp chính thân dày	500m	250m	100m
Chiều dài tối đa của cáp chính thân mỏng	100m	100m	100m
Chiều dài tối đa cáp nhánh	6m	6m	6m

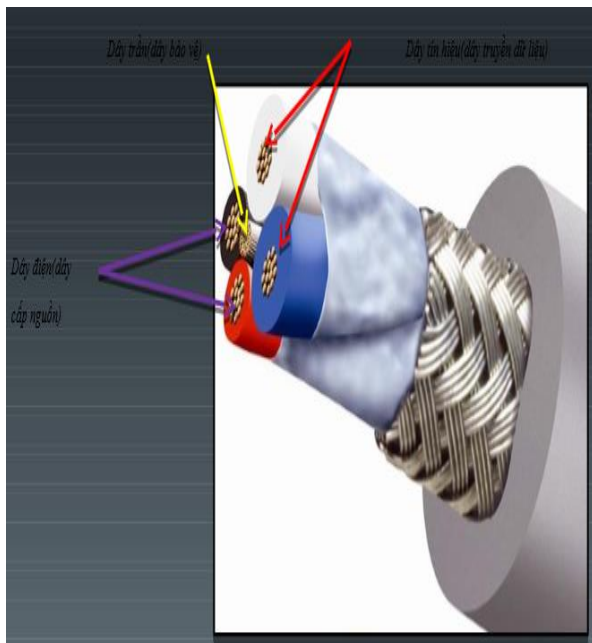
Bảng mối quan hệ giữa tốc độ và chiều dài của cáp



Mô hình một mạng Devicenet

Cấu trúc mạng và kĩ thuật truyền dẫn

- DeviceNet chỉ sử dụng một sợi dây cáp. Dây này vừa là dây nguồn vừa là dây truyền dữ liệu, cáp này thường có 5 sợi

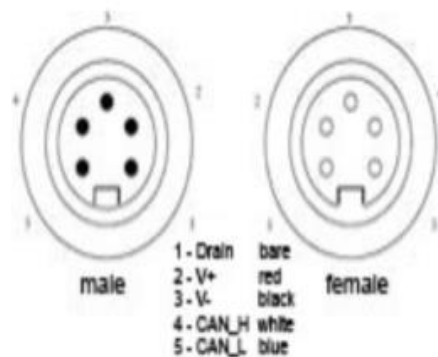


Dây màu	Tín hiệu	Cáp tròn	Cáp dẹt
Trắng	CAN_H	DN tín hiệu	Tín hiệu DN DN
Xanh	CAN_L	DN tín hiệu	DN tín hiệu
Dây trần	Dòng máng	Bảo vệ	Không sử dụng
Đen	V-	Điện	Điện
Đỏ	V-	Điện	Điện

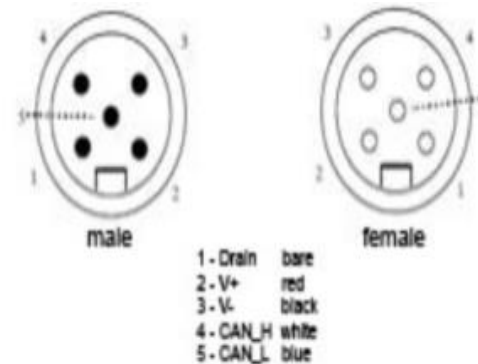
Hình vẽ cắt ngang của một sợi cáp tròn

Cấu trúc mạng và kĩ thuật truyền dẫn

- Đầu nối: Có 3 loại kết nối cơ bản: mở, mini-sealed và micro-sealed. Tùy thuộc vào ứng dụng và các đặc tính của thiết bị mà ta sẽ lựa chọn từng cách ghép nối sao cho phù hợp



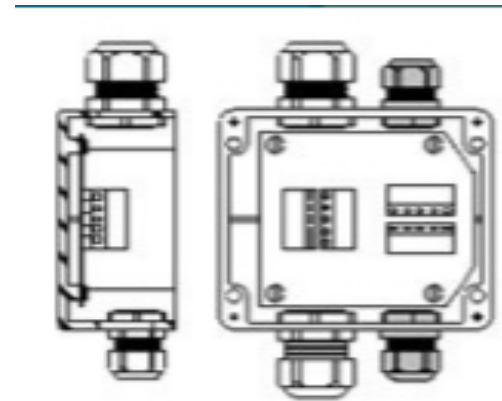
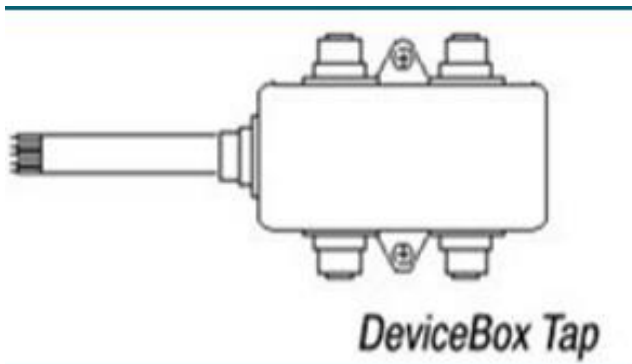
Hình 3: Mini-sealed connector



Hình 4: Micro-sealed connector

Cấu trúc mạng và kĩ thuật truyền dẫn

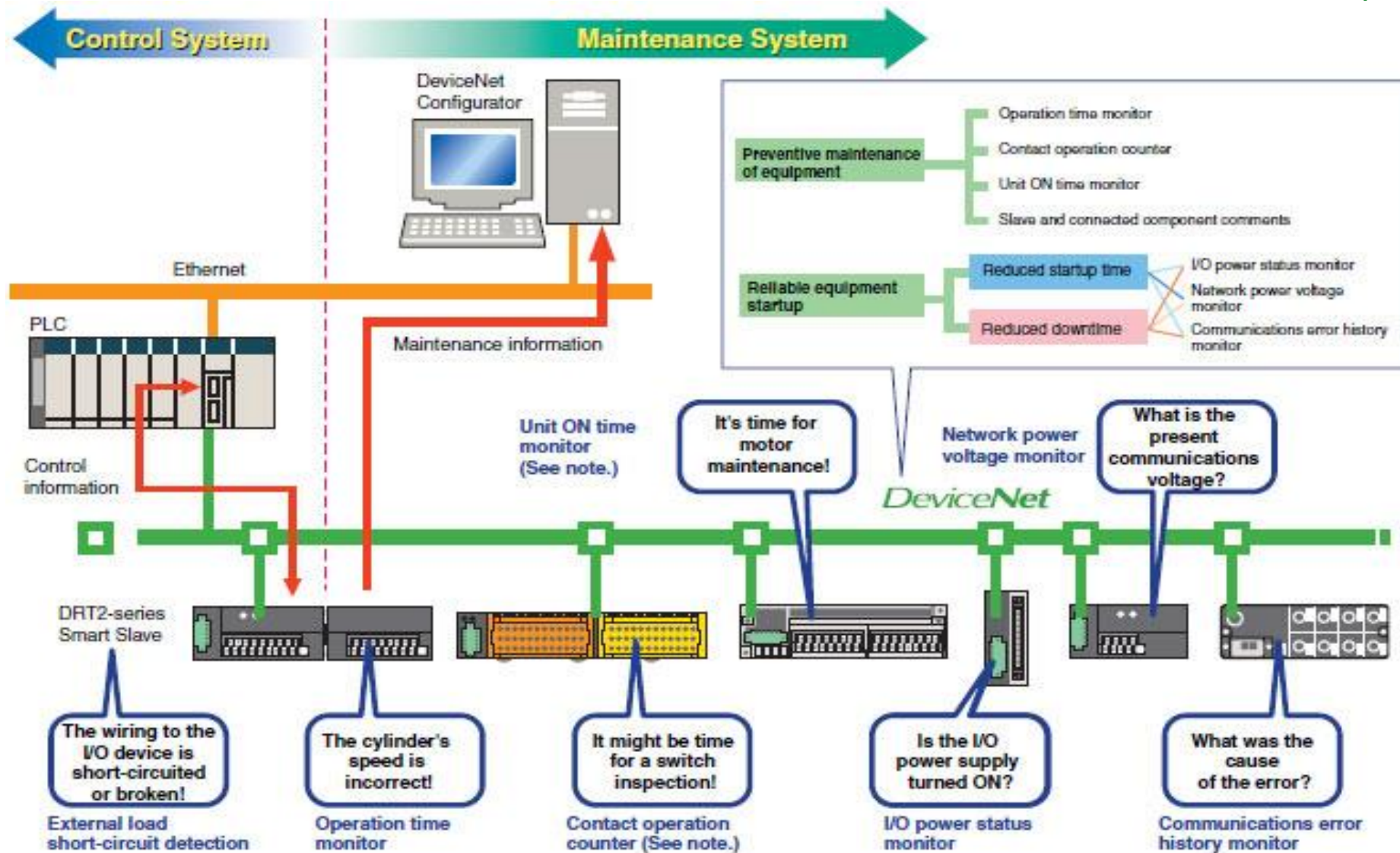
- Ngoài ra còn có các thiết bị khác như là DeviceBox Tap, DeviceBox để kết nối giữa cáp nhánh với cáp chính, giữa cáp chính với cáp chính.



Cơ chế giao tiếp

- Một mạng DeviceNet hoạt động dựa trên mô hình nhà sản xuất/ người tiêu dùng (Producer/Consumer). Trong các bài toán điều khiển, mô hình này cho phép các hình thức giao tiếp sau:
 - ❖ Điều khiển theo sự kiện: một thiết bị chỉ gửi dữ liệu mỗi khi dữ liệu có thay đổi
 - ❖ Điều khiển theo thời gian: Mỗi thiết bị có thể gửi dữ liệu một cách tuần hoàn theo chu kì do người sử dụng đặt
 - ❖ Gửi đồng loạt: thông báo được gửi đồng thời tới tất cả hoặc một nhóm thiết bị
 - ❖ Hỏi tuần tự: Phương pháp cổ điển cho các hệ thống có cấu hình chủ /tớ.

Cơ chế giao tiếp



Cấu trúc bức điện

- Khung bức điện DeviceNet được mô tả ở trên hình vẽ, trường thông tin dữ liệu nhỏ hơn 8 byte, khi truyền các bức điện lớn ta phải phân mảnh dữ liệu.

1 bit	11 bits	1 bit	6 bits	0-8 bytes	15 bits	1 bit	1 bit	1 bit	7 bits	3 bits
Start of Frame	Identifier	RTR bit	Control Field	Data Field (0...8 bytes)	CRC Sequence	CRC Delimiter	Ack Slot	Ack Delimiter	End of Frame	Interframe Space
Arbitration Field										

Mô hình cấu trúc bức điện

Cấu trúc bức điện

Nó bao gồm các phần như sau:

- Khởi đầu khung (Start of Frame): là một bit trội và đánh dấu khởi đầu của một khung dữ liệu hoặc khung yêu cầu dữ liệu. Tất cả các trạm sẽ phải đồng bộ hóa dựa vào bit khởi đầu này.
- Mã căn cước (Identifier): dài 11 bit, nó không nói lên địa chỉ đích của thông báo, mà chỉ biểu diễn ý nghĩa của dữ liệu trong thông báo. Do đó mỗi trạm trên mạng có thể tự quyết định tiếp nhận và xử lý thông báo hay không tiếp nhận thông báo.
- Bit RTR (Remote Transmission Request): dùng để phân biệt giữa khung dữ liệu (bit trội) và khung yêu cầu dữ liệu (bit lặn).

Cấu trúc bức điện

- Ô điều khiển (Control Field): dài 6 bit, trong đó có 4 bit cuối mã hóa chiều dài dữ liệu (bit trội = 0, bit lặn = 1). Tùy theo dạng khung là chuẩn hay mở rộng mà ý nghĩa của hai bit còn lại khác nhau một chút.
- Ô dữ liệu (Data Field): có chiều dài từ 0-8 byte, trong đó mỗi byte được truyền đi theo thứ tự từ bit có giá trị cao nhất (MSB) đến bit có giá trị thấp nhất (LSB).
- Ô kiểm soát lỗi CRC (CRC Sequence): bao gồm 15 bit được tính theo phương pháp CRC và 1 bit lặn phân cách (CRC Delimiter). Dãy bit đầu vào để tính bao gồm bit khởi đầu khung, mã căn cước, ô điều khiển và ô dữ liệu.

Cấu trúc bức điện

- Ô xác nhận ACK (Acknow legment) : gồm 2 bit, được phát đi là các bit lặn. Mỗi trạm nhận được bức điện phải kiểm tra mã CRC, nếu đúng sẽ phát chồng một bit trội trong thời gian nhận được bit ARC đầu tiên (ARC slot).
- Khung kết thúc (End of Frame) : được đánh dấu bằng 7 bit lặn.
- Để phân biệt các khung bức điện với nhau thì ta dùng một khoảng cách ít nhất là 3 bit lặn, được gọi là Interframe Space

Dịch vụ thông báo

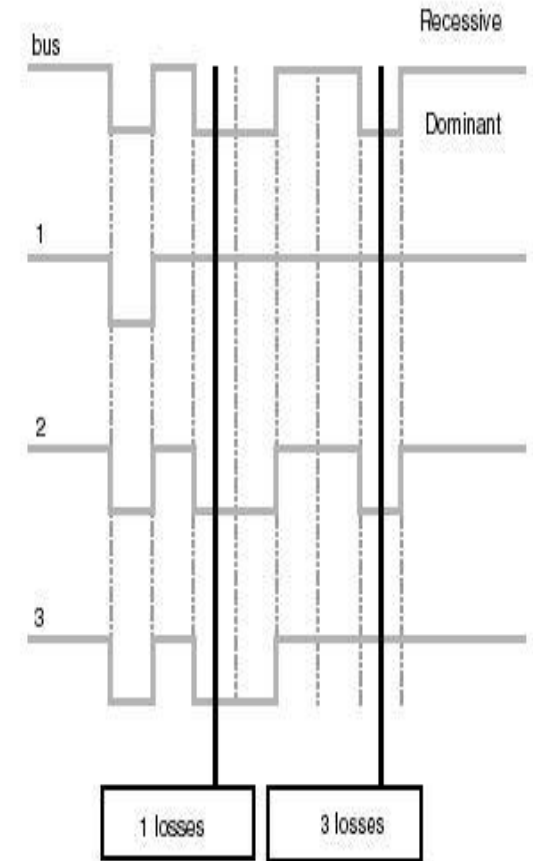
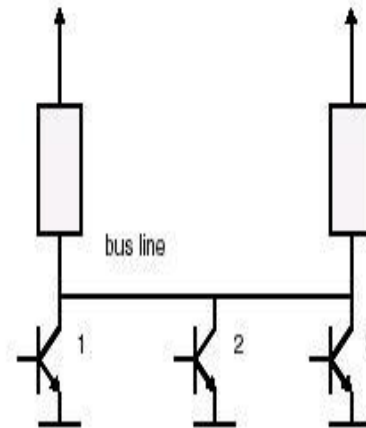
- DeviceNet phân biệt hai kiểu thông báo là thông báo rõ ràng (Explicit Messaging) và thông báo vào/ra (I/O-Messaging).
- Đối với kiểu thông báo rõ ràng, một thông báo mang địa chỉ đầy đủ của thuộc tính cần truy cập hoặc dịch vụ cần gọi. Đây là kiểu giao tiếp có yêu cầu và đáp ứng. Còn các thông báo vào/ra chỉ mang dữ liệu, được tự động gửi đi chứ không nhất thiết phải có yêu cầu.
- Việc trao đổi các thông báo vào/ra thường được thực hiện trong cấu hình giao tiếp chủ/tớ, với các phương pháp như sau:
 - ❖ Polling (Hỏi tuần tự).
 - ❖ Strobing (Quét đồng loạt).
 - ❖ Cyclic (Tuần hoàn).
 - ❖ Change of State (Thay đổi trạng thái).

Truy nhập Bus

- DeviceNet giống như CAN cũng sử dụng phương thức truy nhập bus là CSMA/CA với sự phân xử từng bit. Sự phân xử đó được thực hiện dựa theo từng bit của mã căn cước (Identifier) trong khung bức điện khi hai hoặc nhiều trạm cùng đồng thời bắt đầu gửi thông báo
- Theo quy ước thì bit giá trị 0 ứng với mức trội và bit giá trị 1 ứng với mức lặn, bit 0 sẽ lấn át. Vì vậy, thông báo nào có mã căn cước càng bé thì mức ưu tiên càng cao

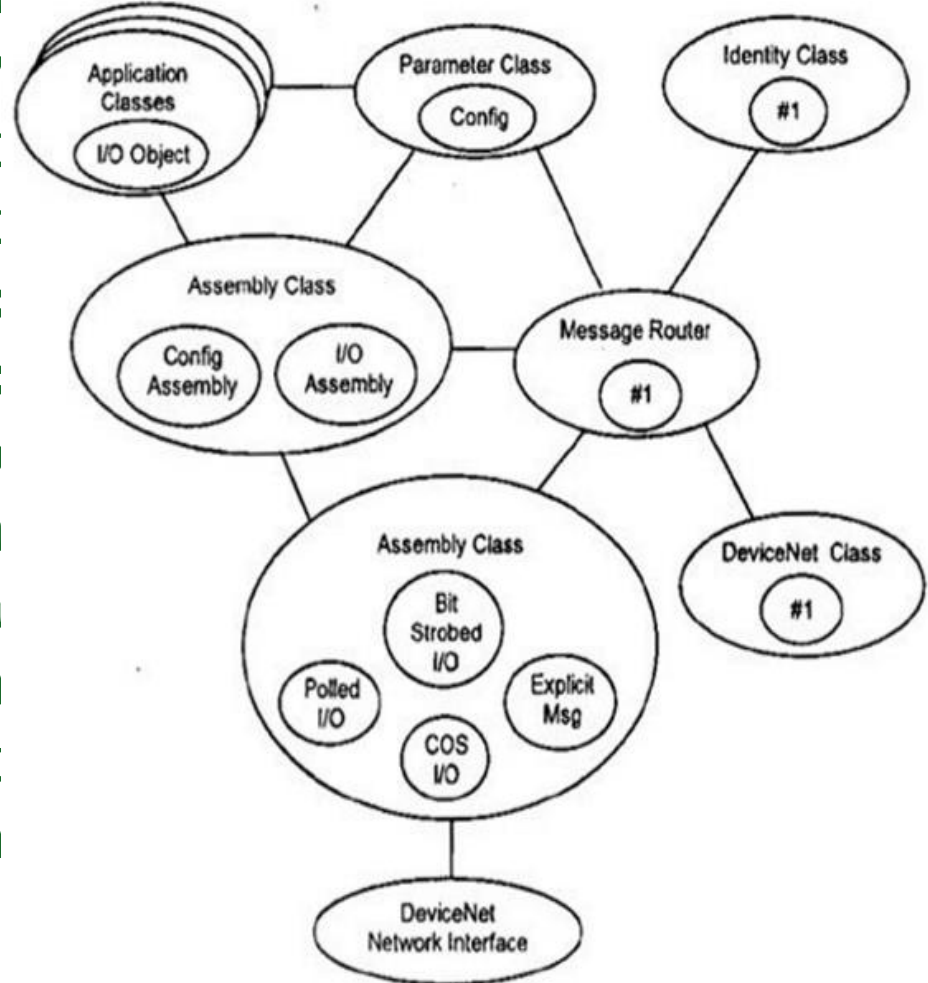
Truy nhập Bus

- Trên hình vẽ mô tả việc truyền dữ liệu đồng thời của nút 1 và 2. Mọi tín hiệu truyền đều bình thường ở vài bit đầu tiên. Khi có sự sai khác giữa 2 bit truyền thì tín hiệu của nút 2 sẽ lấn át nút 1. Lúc bấy giờ nút 1 sẽ mất quyền ưu tiên và ngừng truyền còn nút 2 tiếp tục truyền



Lớp ứng dụng

- Lớp ứng dụng của DeviceNet được xây dựng trên cơ sở một mô hình đối tượng. Một thiết bị DeviceNet được coi là một sưu tập các đối tượng đại diện cho các thành phần của trạm. Mỗi đối tượng là một thể nghiệm (instance) của một trong các lớp mô tả trên hình bên.



Mô hình đối tượng một trạm thiết bị DeviceNet

Lớp ứng dụng

- Mỗi đối tượng có một tập hợp các thuộc tính và chức năng dịch vụ. Các đối tượng có ý nghĩa cụ thể như sau:
- Đối tượng căn cước (Identity Object): Chứa các thuộc tính như mã số nhà sản xuất (Vendor ID), kiểu thiết bị (Device Type). Phiên bản (Revision), trạng thái (Status), số serial (Serial Number) và tên sản phẩm (Product Name)

Lớp ứng dụng

- Đối tượng chuyển thông báo (Message Router Object): Chuyển tiếp thông báo tới các đối tượng khác, thông thường không chứa các thuộc tính nào có thể truy cập mạng.
- Đối tượng DeviceNet (DeviceNet Object): Chứa các thuộc tính như địa chỉ trạm (MAC-ID), tốc độ truyền, hành động khi ngắt bus (Bus-Off Action), số đếm lần ngắt bus (Bus-Off Counter) và địa chỉ trạm chủ (Master's MAC-ID).

Lớp ứng dụng

- Đối tượng ghép (Assembly Object): Đối tượng tùy chọn này tổng hợp thuộc tính của nhiều đối tượng ứng dụng khác nhau, để có thể gửi đồng loạt cho chúng một thông báo duy nhất.
- Đối tượng nối (Connection Object): Đại diện một điểm cuối của một đường nối ảo giữa hai trạm của một mạng.
- Đối tượng tham số (Parameter Object): Đối tượng tùy chọn này đóng vai trò giao diện dữ liệu cấu hình của một thiết bị. Các thuộc tính bao gồm giá trị (Value), phạm vi (Ranges), chuỗi (Strings) và giới hạn (Limits).
- Đối tượng ứng dụng (Application Object): Đại diện cho chính chương trình ứng dụng.

Lớp ứng dụng

- Mỗi đối tượng chứa một số thuộc tính có thể và thực hiện một số dịch vụ có thể khai thác được trên mạng.
- Việc truy nhập mỗi thuộc tính hoặc sử dụng mỗi dịch vụ của một đối tượng thông qua một địa chỉ phân biệt.
- Mỗi địa chỉ này được cấu thành bởi địa chỉ trạm (MAC-ID), mã căn cước lớp đối tượng (Object Class Identifier), mã số đối tượng (Instance Number) và mã số thuộc tính/dịch vụ.

Lớp ứng dụng

- Nguyên tắc định địa chỉ thuộc tính và dịch vụ được minh họa trên hình bên. Khoảng giá trị cho các thành phần địa chỉ được quy định như sau:

- ❖ Địa chỉ MAC (MAC-ID): 0..63
- ❖ Căn cước lớp đối tượng (Object Class Identifier): 1..65535
- ❖ Số thứ tự thể nghiệm (Instance Number): 0..65535
- ❖ Số thứ tự thuộc tính (Attribute Number): 1..255

