

# CAN/CANbus là gì?

Controller Area Network (CAN hoặc CAN Bus) là công nghệ mạng nối tiếp, tốc độ cao, bán song công, hai dây. Ban đầu CAN được thiết kế dành cho ngành công nghiệp ô tô, tuy nhiên hiện nay CAN cũng đã trở thành một tiêu chuẩn phổ biến trong tự động hóa công nghiệp và các ngành khác.

## Ưu điểm của CAN

- Đơn giản, chi phí thấp: bus CAN chỉ có 2 dây giúp kết nối các module điều khiển với nhau dễ dàng hơn khi so sánh với cách làm truyền thống. Kèm theo đó là nhiều lợi ích về việc dễ lắp đặt và dễ sửa chữa, bảo trì khi có sự cố.
- Tạo ra một giao thức chung để nhiều nhà cung cấp khác nhau có thể phát triển các module điều khiển tương thích với nhau
- Tính ưu tiên của thông điệp (Prioritization of messages): mỗi thông điệp được truyền ra từ một nút (node) hay trạm (station) trên bus CAN đều có mức ưu tiên. Khi nhiều thông điệp được truyền ra bus cùng lúc thì thông điệp có mức ưu tiên cao nhất sẽ được truyền. Các thông điệp có mức ưu tiên thấp hơn sẽ tạm dừng và được truyền lại khi bus rảnh. Việc xác định mức ưu tiên của thông điệp dựa trên cấu tạo (cấu trúc) thông điệp và cơ chế phân xử quy định trong chuẩn chuẩn CAN.
- Cấu hình linh hoạt: cho phép thiết lập cấu hình thời gian bit, thời gian đồng bộ, độ dài dữ liệu truyền, dữ liệu nhận, ...
- Nhận dữ liệu đa điểm với sự đồng bộ thời gian: một thông điệp có thể được nhận bởi nhiều node khác nhau trong bus cùng lúc. Tất cả các node trên bus đều có thể thấy thông điệp đang truyền trên bus, tùy vào cấu hình ở mỗi node mà node sẽ quyết định có chấp nhận thông điệp này hay không.
- Nhiều master (multimaster)
- Phát hiện và báo hiệu lỗi: Mỗi thông điệp có kèm theo mã CRC (Cyclic Redundancy Code) để thực hiện kiểm tra lỗi. Nếu lỗi xuất hiện, node nhận sẽ bỏ qua thông điệp lỗi và truyền khung báo lỗi (error frame) lên bus CAN. Mỗi node trong bus có bộ đếm quản lý lỗi truyền nhận riêng để xác định trạng thái lỗi của chính nó. Nếu lỗi xuất hiện quá nhiều, một node có thể tự động ngắt khỏi bus. Ngoài ra còn một số dạng lỗi khác có thể được phát hiện với chuẩn CAN.
- Tự động truyền lại các thông điệp bị lỗi khi bus rảnh: Một thông điệp được truyền ra bus nếu bị lỗi thì sẽ không mất đi mà node truyền thông điệp này sẽ giữ nó lại và tự động phát lại thông điệp này khi bus CAN rảnh cho đến khi thành công. Điều này giúp đảm bảo tính toàn vẹn dữ liệu trong bus

Những đặc điểm trên, bạn đọc có thể thấy rõ khi đi sâu vào nghiên cứu chuẩn này thông qua thông tin chia sẻ bên dưới:

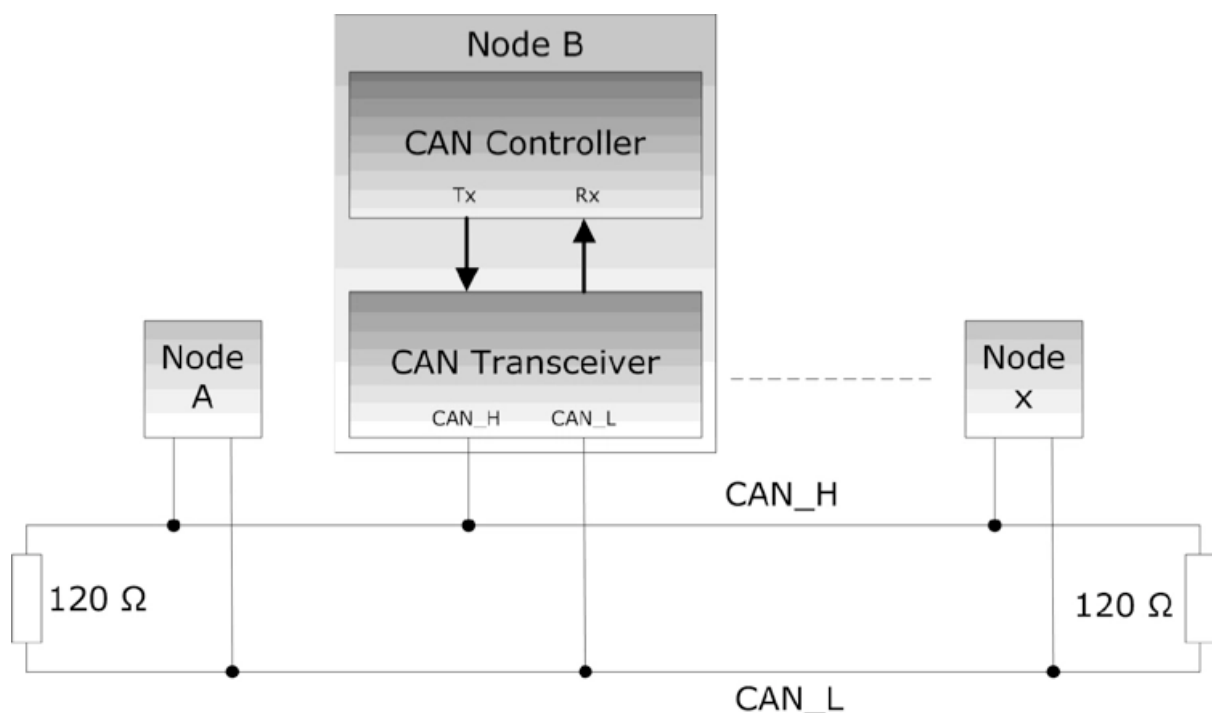
## CAN hoạt động như thế nào?

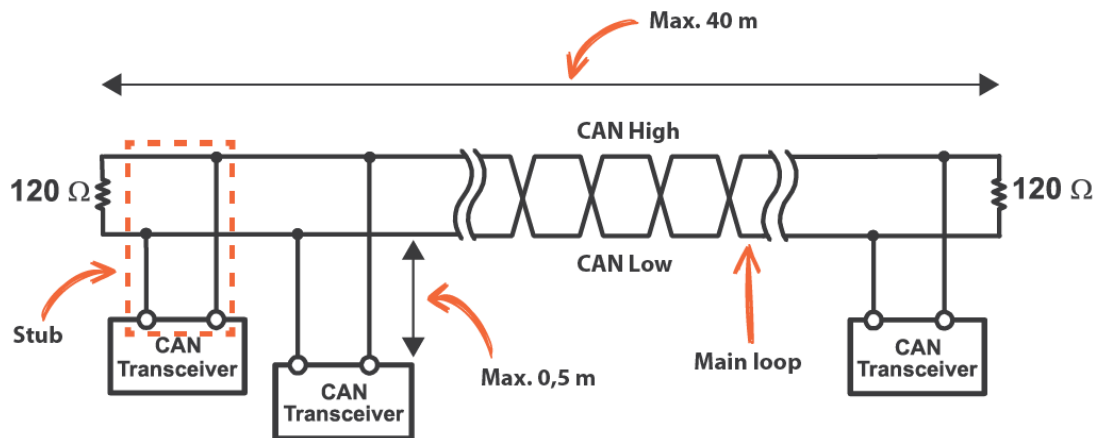
CAN sử dụng CSMA / CD + AMP (Carrier Sense Multiple Access / Collision Detection with Arbitration on Message Priority). Trước khi gửi thông điệp, nút CAN sẽ kiểm tra xem bus có bận không. Nó cũng sử dụng để phát hiện khả năng trùng lặp. Những phương thức này cũng tương tự như [Ethernet](#). Tuy nhiên, khi một mạng Ethernet phát hiện xung đột, cả hai nút gửi sẽ ngừng truyền. Sau đó, nó đợi một khoảng thời gian trễ ngẫu nhiên trước khi thử gửi lại. Điều này làm cho mạng Ethernet rất nhạy cảm với tải bus cao. Có thể giải quyết vấn đề này với nguyên tắc xác định quyền ưu tiên rất thông minh trong mạng CAN.

### Nguyên tắc

Thông điệp dữ liệu được truyền từ bất kỳ nút nào trên bus CAN không chứa địa chỉ của nút truyền hoặc của bất kỳ nút nhận dự kiến nào.

Thay vào đó, nội dung của thông điệp được gắn nhãn bởi một số nhận dạng (ID) là duy nhất trên toàn mạng. Tất cả các nút khác trên mạng đều nhận được thông điệp và mỗi nút thực hiện kiểm tra sự chấp nhận trên mã ID để xác định xem thông điệp có liên quan đến nút đó hay không. Nếu thông điệp có liên quan, nó sẽ được xử lý; nếu không thì nó bị bỏ qua.





## Các loại CAN Frame

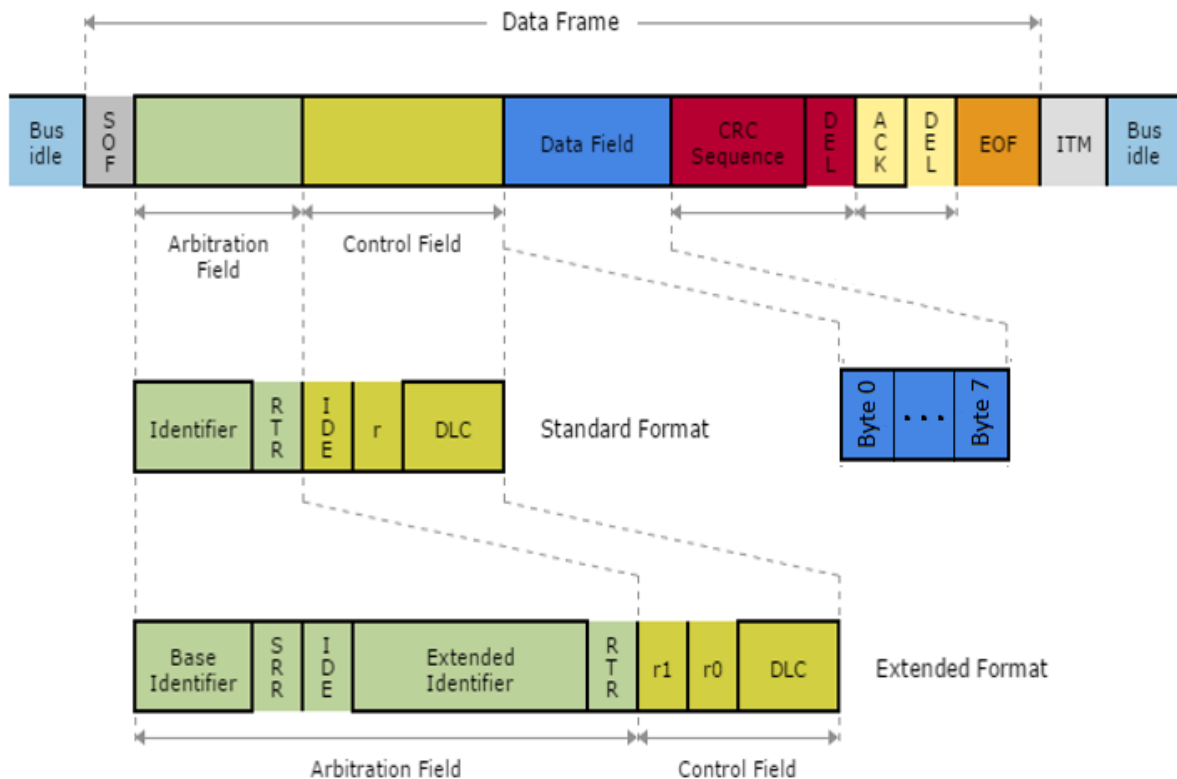
Dữ liệu CAN được truyền dưới dạng các Frame (khung). Có 4 loại Frame khác nhau, đó là:

1. Data Frame (khung dữ liệu): là khung mang dữ liệu từ một bộ truyền dữ liệu đến các bộ nhận dữ liệu. Khung này có vùng để mang các byte dữ liệu.
2. Remote Frame (khung yêu cầu hay điều khiển): là khung được truyền từ một Node bất kỳ để yêu cầu dữ liệu từ Node khác. Khi Node khác đó nhận được yêu cầu sẽ truyền lại dữ liệu có ID (Identifier) trùng với ID được gửi trong Remote Frame.
3. Error Frame (khung lỗi): là khung được truyền bởi bất kỳ Node nào khi Node đó phát hiện lỗi từ Bus.
4. Overflow Frame (khung báo tràn): mỗi Node trong CAN Bus có thể truyền bất kỳ khi nào nếu phát hiện Bus rảnh. Hoặc nếu một Node nhận quá nhiều dữ liệu và không xử lý kịp, nó sẽ gửi Frame này để các Node khác không gửi thêm dữ liệu cho nó.

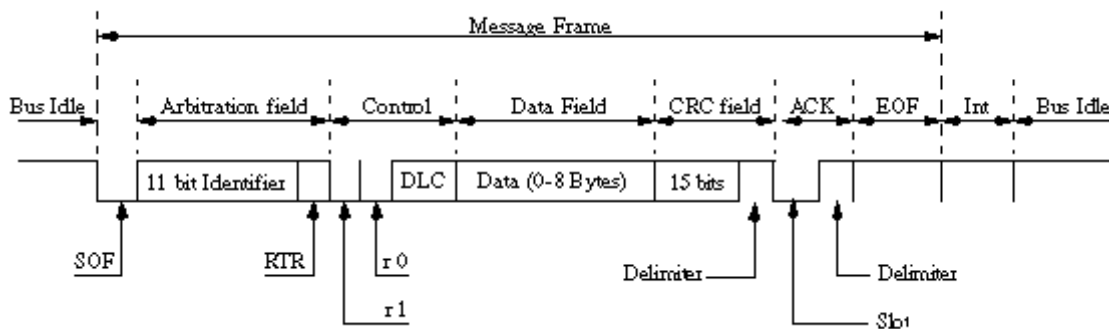
Data Frame và Remote Frame làm việc theo cơ chế phân xử quyền ưu tiên của tín hiệu vì thế cấu trúc của chúng có vùng phân xử quyền ưu tiên, nơi chứa ID của khung. Và chúng có hai định dạng khác nhau là định dạng chuẩn (Standard) và định dạng mở rộng (Extended):

- Định dạng khung chuẩn sử dụng ID có độ dài 11 bit
- Định dạng khung mở rộng sử dụng ID có độ dài 29 bit

## Data Frame (khung dữ liệu)



## CAN 2.0A



Data Frame CAN (Phiên bản 2.0A) tiêu chuẩn bao gồm bảy trường bit khác nhau:

1. Trường bắt đầu khung (Start Of Frame Field – SOF)

Với cả 2 định dạng của chuẩn CAN 2.0 thì trường bắt đầu là vị trí của bit đầu tiên trong khung. Trường này chiếm 1 bit dữ liệu. Bit đầu tiên này là một Dominant Bit (mức logic 0) đánh dấu sự bắt đầu của một Data Frame.

## 2. Trường xác định quyền ưu tiên (Arbitration Field)

Định dạng vùng xác định quyền ưu tiên là khác nhau đối với dạng khung chuẩn và khung mở rộng.

- Định dạng chuẩn: vùng xác định quyền ưu tiên có độ dài 12 bit, bao gồm 11 bit ID và 1 bit RTR.
- Định dạng mở rộng: trường xác định quyền ưu tiên có độ dài 32 bit, bao gồm có 29 bit ID, 1 bit SRR, 1 bit IDE và 1 bit RTR

Trong đó:

### Bit RTR (Remote Transmission Request)

- Là bit dùng để phân biệt khung là Data Frame hay Remote Frame.
- Nếu là Data Frame, bit này luôn bằng 0 (Dominant Bit).
- Nếu là Remote Frame, bit này luôn bằng 1 (Recessive Bit).
- Vị trí bit này luôn nằm sau bit ID.

Trường hợp nếu Data Frame và Remote Frame có cùng ID được gửi đi đồng thời thì Data Frame sẽ được ưu tiên hơn.

### Bit SRR (Substitute Remote Request)

- Bit này chỉ có ở khung mở rộng.
- Bit này có giá trị là 1 (Recessive Bit).
- So với vị trí tương ứng trong khung chuẩn thì bit này trùng với vị trí của bit RTR nên còn được gọi là bit thay thế (Substitute).

Giả sử có hai Node cùng truyền, một Node truyền Data Frame chuẩn, một Node truyền Data Frame mở rộng có ID giống nhau thì Node truyền khung chuẩn sẽ thắng phân xử quyền ưu tiên vì đến vị trí sau ID, khung chuẩn là bit RTR = 0, còn khung mở rộng là bit SRR = 1. Như vậy, khung chuẩn chiếm ưu thế hơn so với khung mở rộng khi có ID như nhau.

### Bit IDE (Identifier Extension)

- Đây là bit phân biệt giữa loại khung chuẩn và khung mở rộng: IDE = 0 quy định khung chuẩn, IDE = 1 quy định khung mở rộng.
- Bit này nằm ở trường xác định quyền ưu tiên với khung mở rộng và ở trường điều khiển với khung chuẩn.

## 3. Trường điều khiển (Control Field)

Khung chuẩn và khung mở rộng có định dạng khác nhau ở trường này:

- Khung chuẩn gồm IDE, r0 và DLC (Data Length Code).
- Khung mở rộng gồm r1, r0 và DLC.

Trong đó:

Bit IDE

Dùng phân biệt loại khung (đã được trình bày ở trên).

Bit r0, r1 (hai bit dự trữ)

Tuy hai bit này phải được truyền là Recessive Bit bởi bộ truyền nhưng bộ nhận không quan tâm đến giá trị 2 bit này. Bộ nhận có thể nhận được các tổ hợp 00, 01, 10 hoặc 11 của r1 và r0 nhưng không coi đó là lỗi mà bỏ qua và nhận thông điệp bình thường.

DLC (Data Length Code)

- Có độ dài 4 bit quy định số byte của trường dữ liệu của Data Frame
  - Chỉ được mang giá trị từ 0 đến 8 tương ứng với trường dữ liệu có từ 0 đến 8 byte dữ liệu. Data Frame có thể không có byte dữ liệu nào khi DLC = 0.
  - Giá trị lớn hơn 8 không được phép sử dụng. Hình dưới mô tả các loại mã bit mà DLC có thể chứa để quy định số byte của trường dữ liệu.
4. Trường dữ liệu (Data Field)

Trường này có độ dài từ 0 đến 8 byte tùy vào giá trị của DLC ở trường điều khiển.

#### 5. Trường kiểm tra (Cyclic Redundancy Check Field – CRC)

Trường kiểm tra hay trường CRC gồm 16 bit và được chia làm hai phần là:

- CRC Sequence: gồm 15 bit CRC tuần tự
- CRC Delimiter: là một Recessive Bit làm nhiệm vụ phân cách trường CRC với trường ACK
- Mã kiểm tra CRC phù hợp nhất cho các khung mà chuỗi bit được kiểm tra có chiều dài dưới 127 bit, mã này thích hợp cho việc phát hiện các trường hợp sai nhóm (Bus Error). Ở đây, tổng bit từ trường bắt đầu (SOF) đến trường dữ liệu (Data Field) tối đa là 83 bit (khung định dạng chuẩn) và 103 bit (khung định dạng mở rộng).

=> Trường CRC bảo vệ thông tin trong Data Frame và Remote Frame bằng cách thêm các bit kiểm tra dự phòng ở đầu khung truyền. Ở đầu khung nhận, cũng sẽ tính toán CRC như bộ truyền khi đã nhận dữ liệu và so sánh kết quả đó với CRC Sequence mà nó đã nhận được, nếu khác nhau tức là đã có lỗi, nếu giống nhau tức là đã nhận đúng từ trường SOF đến trường dữ liệu.

#### 6. Trường báo nhận (Acknowledge Field – ACK)

Trường báo nhận hay trường ACK có độ dài 2 bit và bao gồm hai phần là ACK Slot và ACK Delimiter.

- ACK Slot: có độ dài 1 bit, một Node truyền dữ liệu sẽ thiết lập bit này là Recessive. Khi một hoặc nhiều Node nhận chính xác giá trị thông điệp (không có lỗi và đã so sánh CRC Sequence trùng khớp) thì nó sẽ báo lại cho bộ truyền bằng cách truyền ra một Dominant Bit ngay vị trí ACK Slot để ghi đè lên Recessive Bit của bộ truyền.
- ACK Delimiter: có độ dài 1 bit, nó luôn là một Recessive Bit. Như vậy, ta thấy rằng ACK Slot luôn được đặt giữa hai Recessive Bit là CRC Delimiter và ACK Delimiter.

#### 7. Trường kết thúc (End Of Frame Field – EOF)

Trường EOF là trường thông báo kết thúc một Data Frame hay Remote Frame. Trường này gồm 7 Recessive Bit.