

## *CAN (Controller Area Network)*

Mô tả chung về giao thức:

- CAN là một giao thức thông tin nối tiếp, cung cấp hệ điều khiển thời gian thực , phân tán với độ tin cậy cao.
- CAN là một chuẩn của ISO ( ISO11898).
- CAN được phát triển năm 1980 bởi BOSCH.
- Nó ứng dụng trong CN SX ô-tô, máy công cụ, máy đóng bao...

## *CAN (Controller Area Network)*

CAN bao gồm các lớp sau:

- the (CAN-) object layer
- the (CAN-) transfer layer
- the physical layer.

## *CAN (Controller Area Network)*

- Object:

- + Phát hiện các bản tin đã được truyền đi.
  - + Quyết định bản tin nào sẽ được nhận bởi lớp Transfer và được sử dụng.
  - + Cung cấp giao diện tới người dùng và các phần cứng liên quan.
- Người dùng có thể định nghĩa các đối tượng kết nối.

## *CAN (Controller Area Network)*

- Transfer:

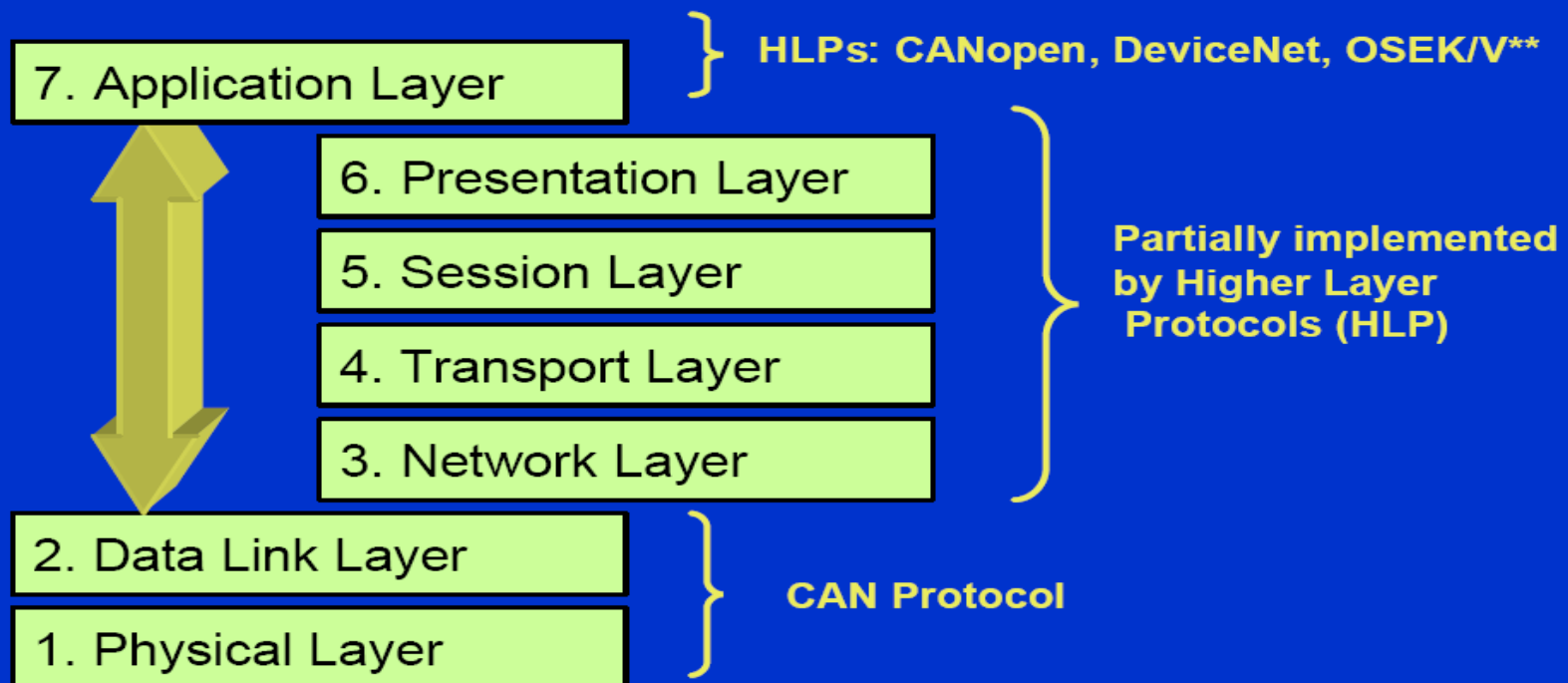
Điều khiển việc định khung, định thời, thực hiện chức năng trọng tài, kiểm tra lỗi, phát hiện lỗi và hạn chế lỗi. Quyết định khi nào thì truyền và nhận bản tin. Người dụng không được tự do thay đổi phần này.

- Physical:

Truyền các bit giữa các nút với 1 tiêu chuẩn về điện.

# CAN (Controller Area Network)

## ISO-OSI\* Reference Model



\*) OSI - Open System Interconnection

CAN

### Layered Structure of a CAN Node

Application Layer
Object Layer <ul style="list-style-type: none"><li>- Message Filtering</li><li>- Message and Status Handling</li></ul>
Transfer Layer <ul style="list-style-type: none"><li>- Fault Confinement</li><li>- Error Detection and Signaling</li><li>- Message Validation</li><li>- Acknowledgement</li><li>- Arbitration</li><li>- Message Framing</li><li>- Transfer Rate and Timing</li></ul>
Physical Layer <ul style="list-style-type: none"><li>- Signal Level and Bit Representation</li><li>- Transmission Medium</li></ul>

## CAN - Các khái niệm cơ bản

- Bản tin: thông tin được truyền theo 1 vài định dạng cố định, có độ dài hạn chế. khi Bus rỗi một thiết bị có thể truyền 1 bản tin.
- Định tuyến thông tin: Một nút CAN không chứa bất kỳ thông tin nào về hệ
  - + Hệ mềm dẻo.
  - + Định tuyến bản tin: sử dụng Identifier và Message Filtering.
- Multi Cast.
- Độ bảo toàn dữ liệu.

## *CAN - Các khái niệm cơ bản*

- Tốc độ truyền.
- Mức độ ưu tiên của bản tin.
- Yêu cầu dữ liệu từ xa.
- Multi Master.
- Trọng tài.



## CAN - Các khái niệm cơ bản

- An toàn dữ liệu:
  - + Kiểm tra lỗi:
    - . Giám sát.
    - .CRC
    - . Nhồi bit.
    - .Kiểm tra khung bản tin.
  - + Hiệu quả:
    - .Tất cả các lỗi toàn cục.
    - .Lỗi tại bộ truyền.
    - . 5 lỗi phân bố ngẫu nhiên.
    - .chuỗi lỗi < 15 bit
    - . Các lỗi lẻ
    - . Còn lại 4.7 10<sup>-11</sup>

## *CAN - Các khái niệm cơ bản*

- Báo lỗi và thời gian phục hồi.
- Hạn chế lỗi.
- Kết nối trong CAN
- Kênh truyền.
- Giá trị bit.
- ACK
- Sleep/Wake up.

## CAN - Cấu trúc bản tin

Các bản tin được truyền và xử lý trong CAN theo 4 kiểu khác nhau của khung bản tin:

- Data Frame: mang thông tin từ nơi phát đến nơi thu.
- Remote Frame: gửi đi một yêu cầu truyền một Data Frame với cùng một Identifier.
- Error Frame: truyền đi bởi bất kỳ một nút nào phát hiện ra lỗi trên Bus.
- Overload Frame: cung cấp một thời gian trễ giữa Data Frame và Remote Frame.

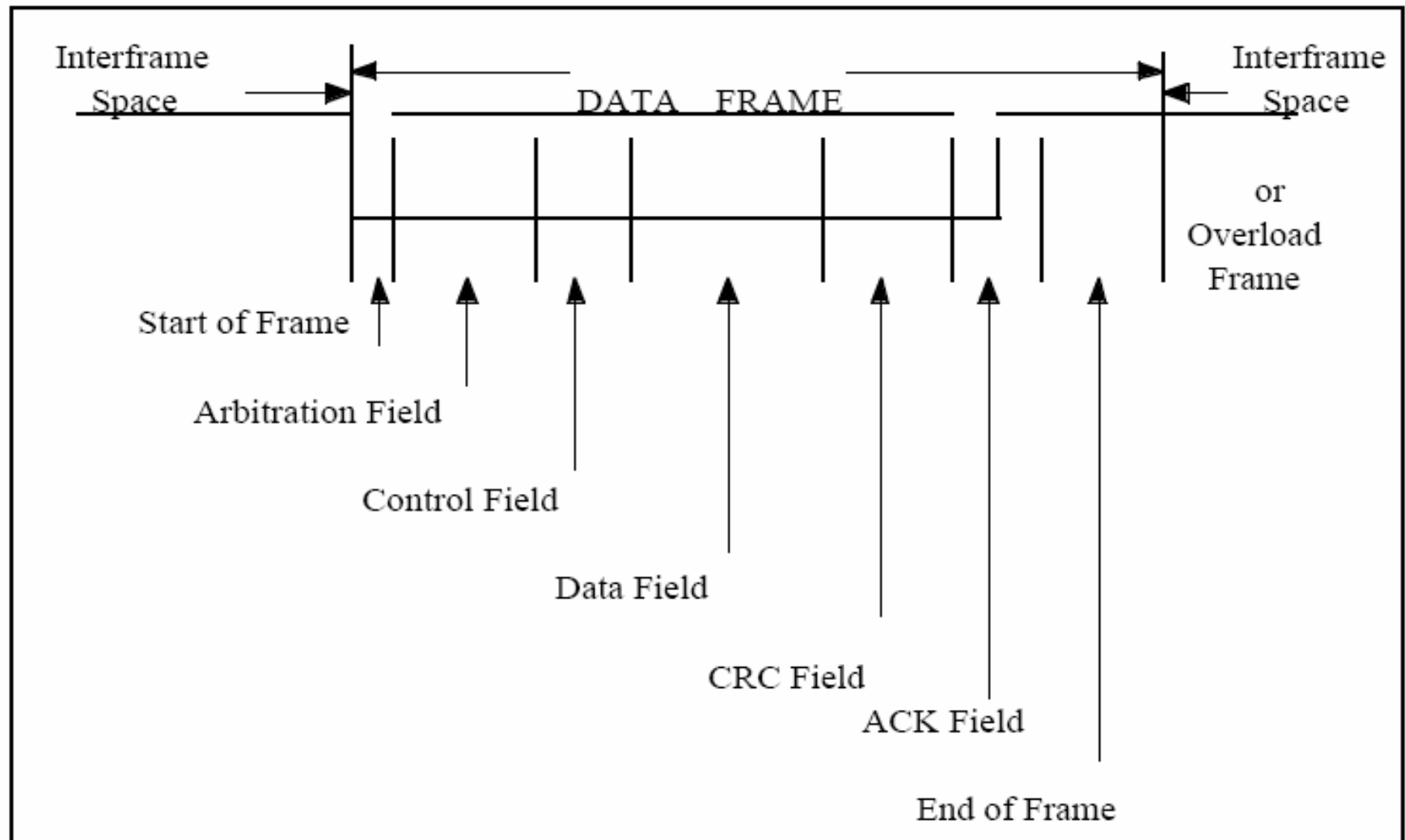
Giữa Data Frame và Remote Frame được phân biệt với nhau bởi InterFrame Space.

## *CAN - Cấu trúc bản tin*

Data Frame: bao gồm 7 trường bit.

- START OF FRAME.
- ARBITRATION FIELD
- CONTROL FIELD
- DATA FIELD
- CRC FIELD
- ACK FIELD
- END OF FRAME

## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

- START OF FRAME: đánh dấu việc bắt đầu một Data Frame, hay Remote Frame. Nó gồm 1 bit "trội" ( Dominant)
- Trạm có thể gửi số liệu khi bus rỗi.
- Các trạm sẽ đồng bộ với sườn của START OF FRAME.

## CAN - Cấu trúc bản tin

ARBITRATION FIELD: bao gồm IDENTIFIER and the RTR-BIT.

IDENTIFIER bao gồm 11 bit ID10 - ID0. Các bit cao truyền đi trước.

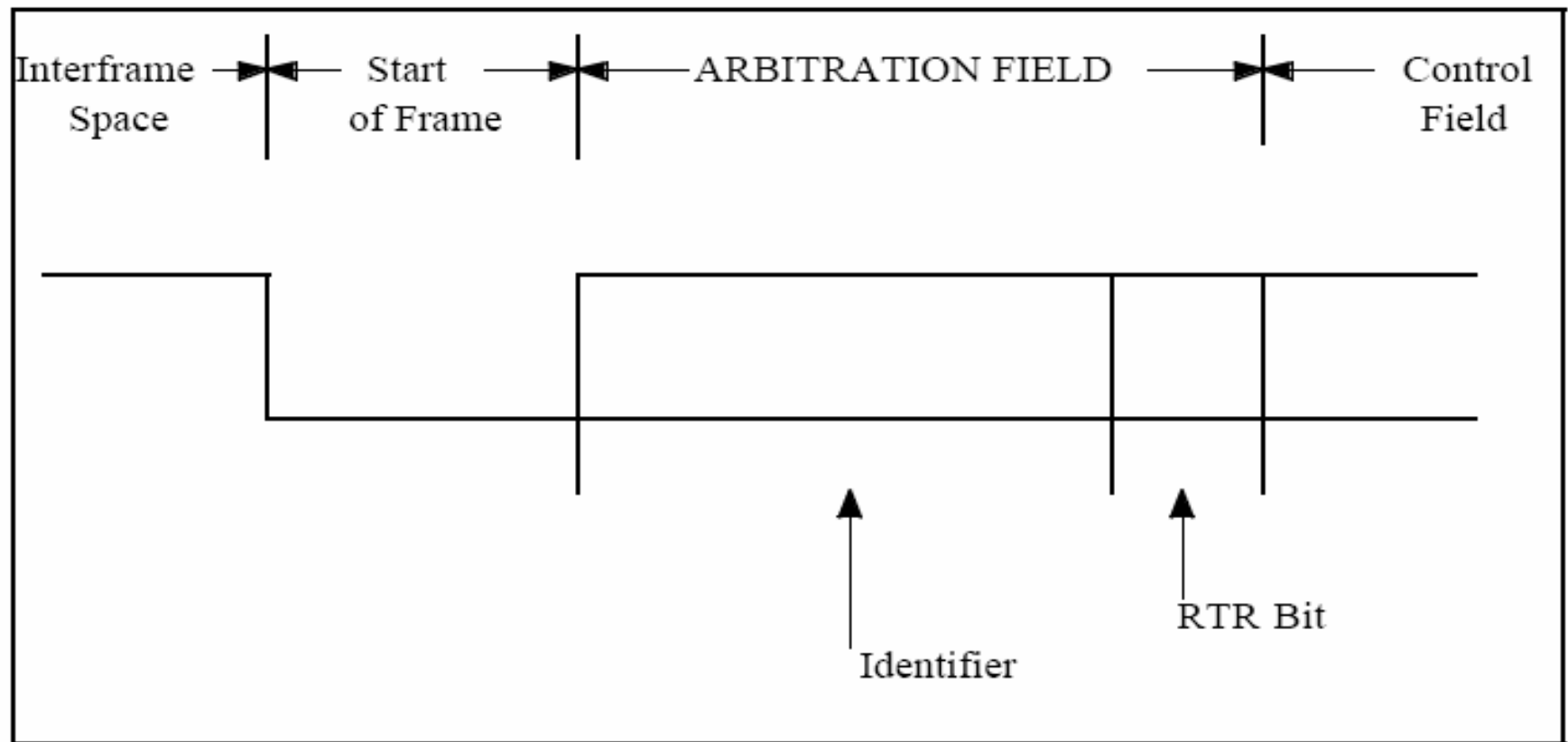
7 bit cao nhất từ ID10-ID4 không được tất cả là "lặn" (Recessive)

RTR BIT (Remote Transmission Request BIT)

"trội" nếu là Data Frame.

"lặn" nếu là Remote Frame.

## CAN - Cấu trúc bản tin





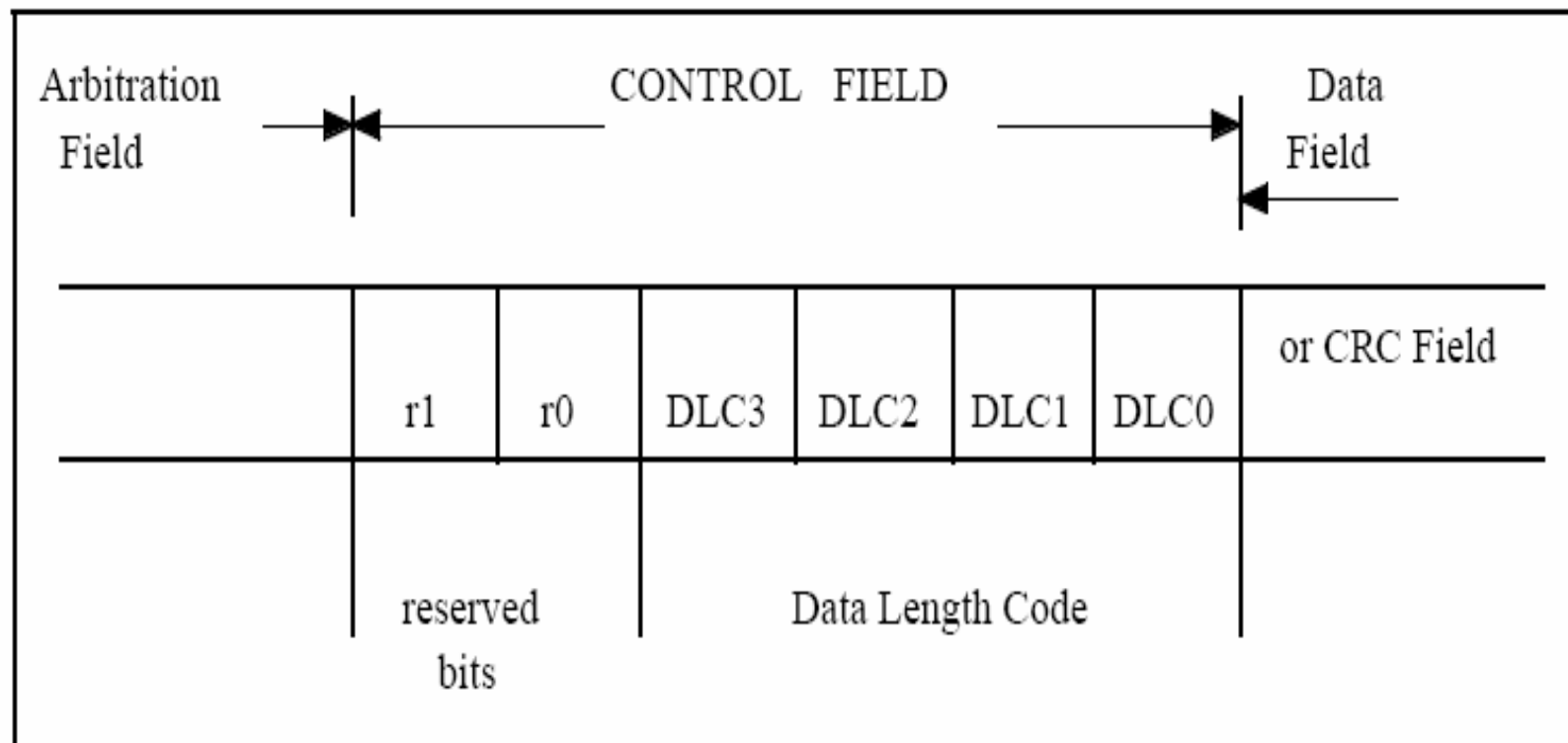
## *CAN - Cấu trúc bản tin*

CONTROL FIELD: bao gồm 6 bit.

4 bit mã hoá độ dài của trường dữ liệu 0...8

2 bit dành cho mở rộng trong tương lai

## CAN - Cấu trúc bản tin



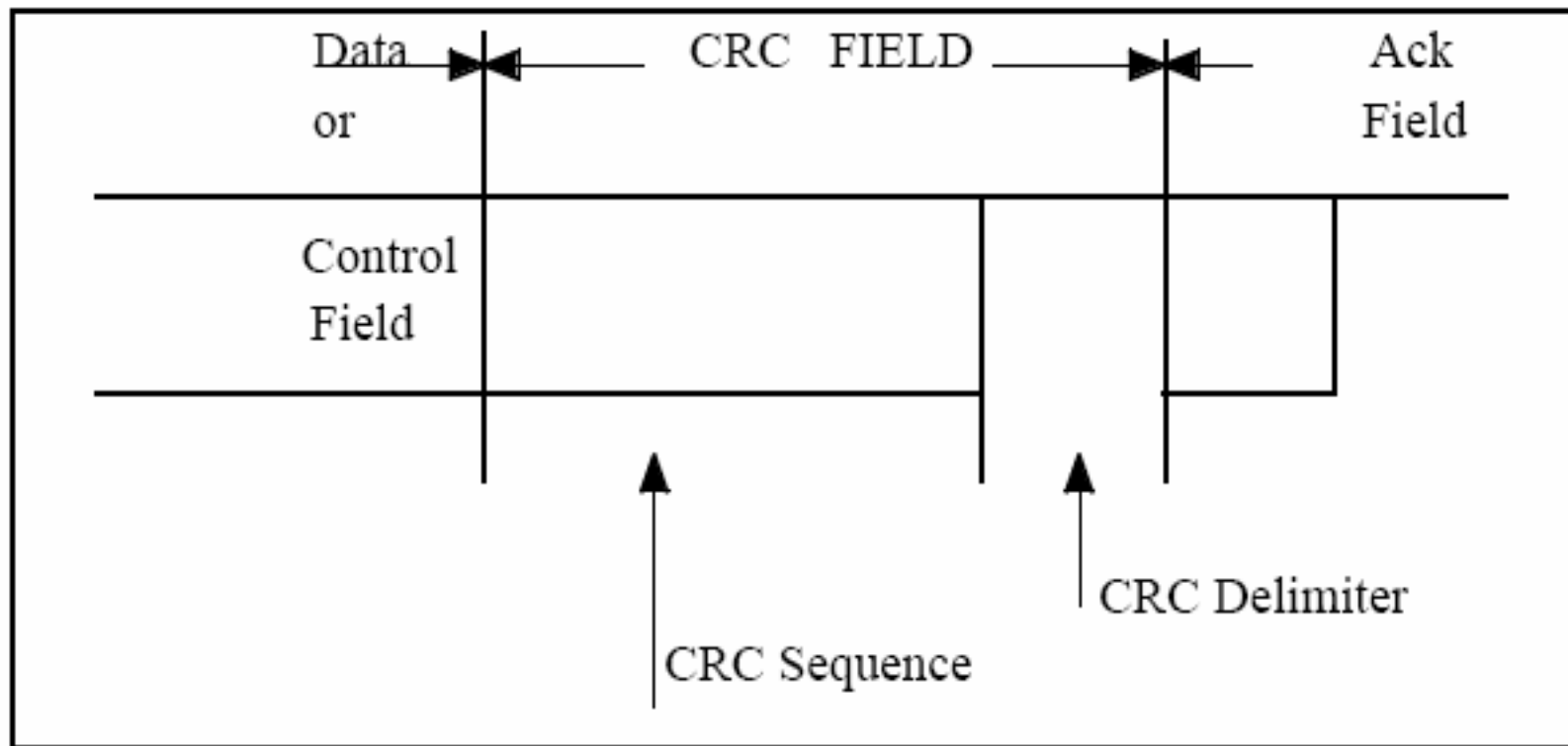
## CAN - Cấu trúc bản tin

Number of Data Bytes	Data Length Code			
	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	d	d

## CAN - Cấu trúc bản tin

- Data Field : Chứa 0...8 byte dữ liệu.
- CRC Field: bao gồm CRC sequence và CRC delimiter  
CRC sequence là kết quả tính toán theo phương pháp CRC các trường: START OF FRAME, ARBITRATION FIELD, CONTROL FIELD, DATA FIELD
- Sử dụng đa thức:  $X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1$ .
- CRC delimiter bao gồm một bit lặn.

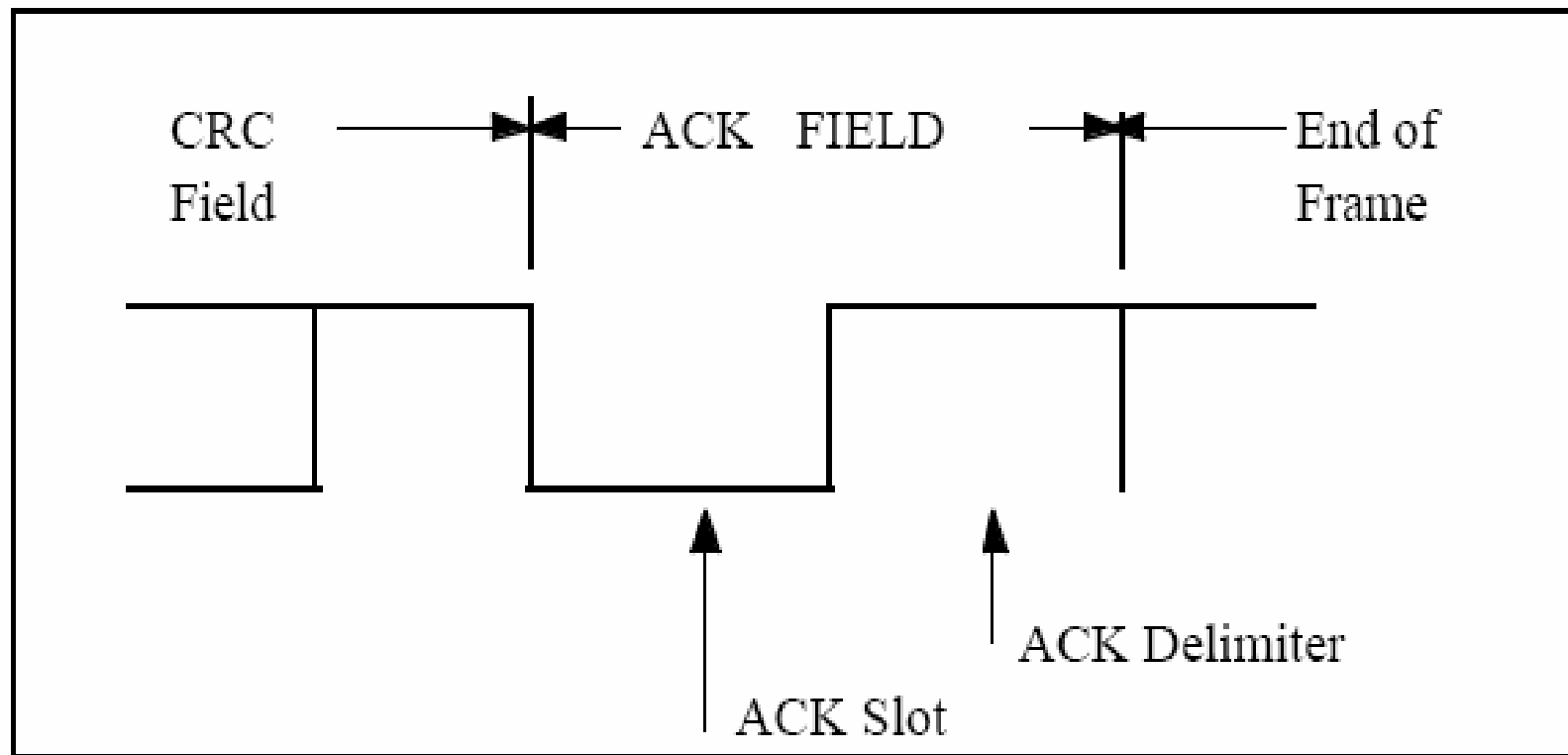
## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

- ACK FIELD bao gồm 2 bit ACK SLOT và ACK DELIMITER.
- Khi truyền bên truyền sẽ gửi đi hai bit lặn.
- Nếu bên nhận nhận tốt bản tin sẽ gửi bit trội vào ACK SLOT
- End of Frame: là một cờ bao gồm 7 bit lặn.

## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

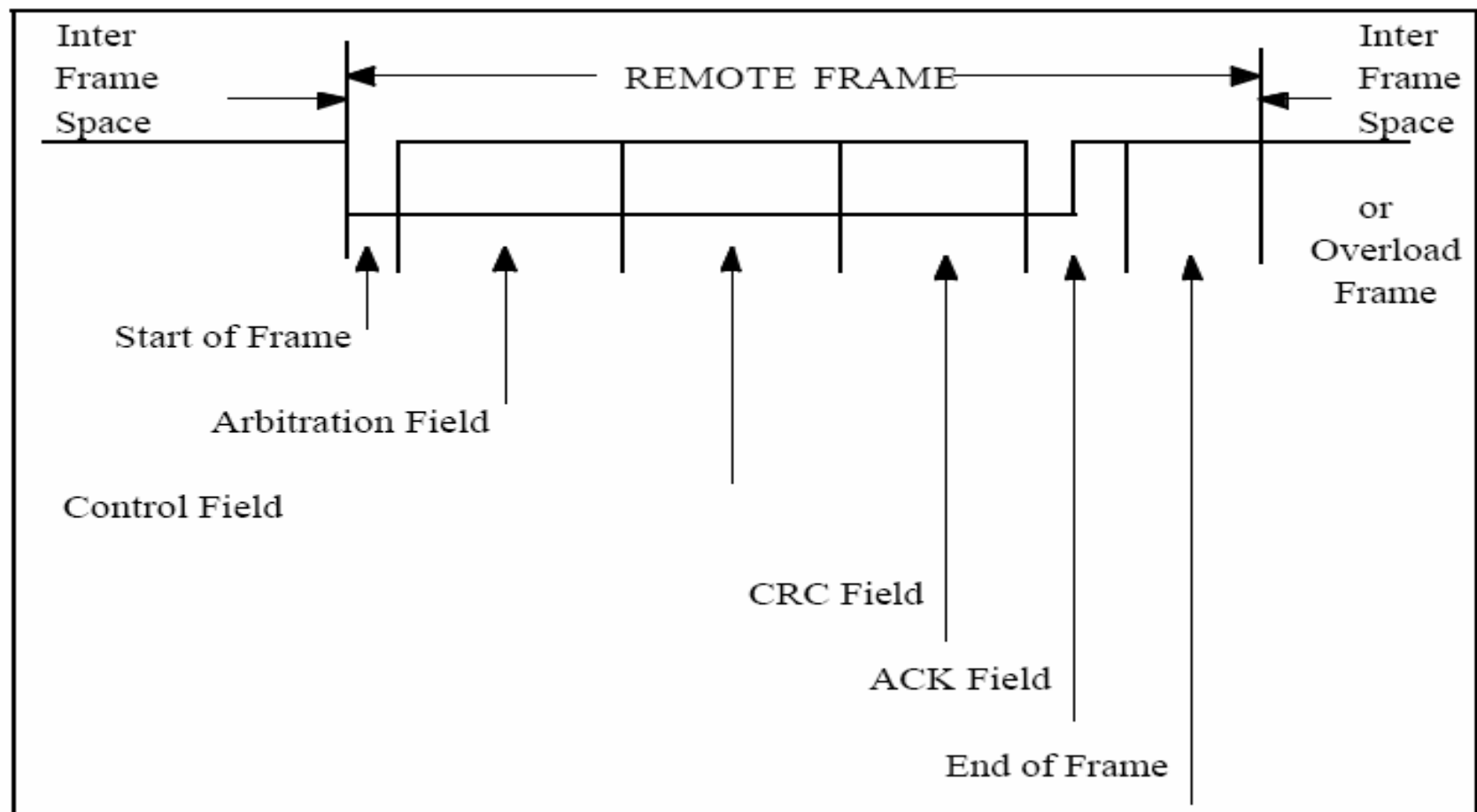
Remote Frame: bao gồm 6 trường bit.

- START OF FRAME.
- ARBITRATION FIELD
- CONTROL FIELD
- CRC FIELD
- ACK FIELD
- END OF FRAME

( Tương tự như Data Frame)



## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

Error Frame bao gồm Error Flag + Error Delimiter.

Error Flag : bao gồm hai loại

ACTIVE ERROR FLAG bao gồm 6 bit trội

PASSIVE ERROR FLAG bao gồm 6 bit lặn

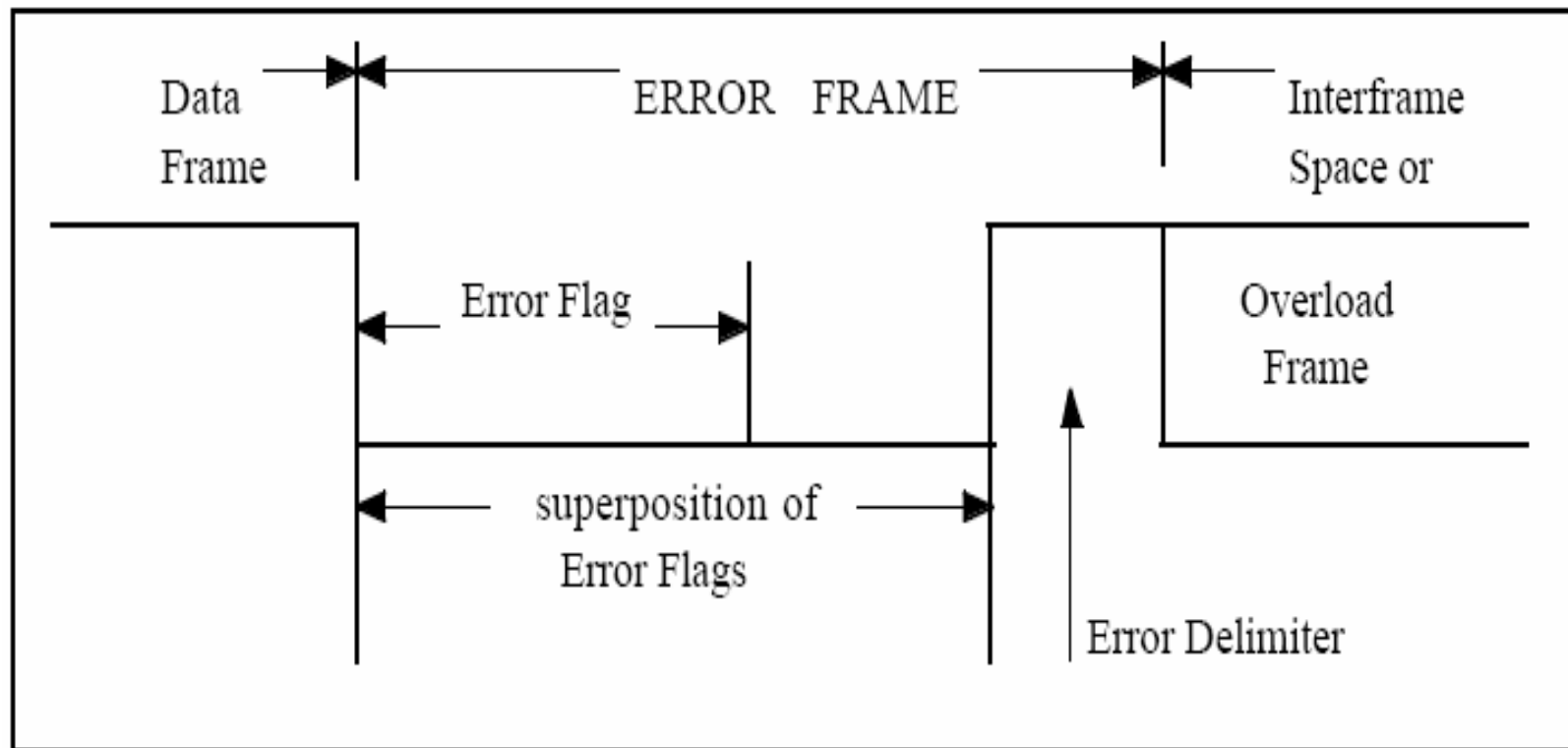
- Một trạm ở trạng thái lỗi tích cực nếu phát hiện ra lỗi sẽ truyền đi một cờ lỗi tích cực. Luật của cờ lỗi sẽ phá huỷ luật chèn bit hay định dạng cố định của bản tin.
- Một trạm ở trạng thái lỗi bị động nếu phát hiện ra lỗi thì sẽ truyền đi cờ lỗi bị động.

## CAN - Cấu trúc bản tin

- ERROR DELIMITER: bao gồm 8 bit lặn.

Sau khi truyền đi cờ lỗi trạm sẽ truyền đi các bit lặn sau đó sẽ giám sát Bus cho tới khi nhận được các bit lặn thì truyền thêm 7 hay nhiều hơn các bit lặn.

## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

OVERLOAD FRAME bao gồm hai trường OVERLOAD FLAG và OVERLOAD DELIMITER.

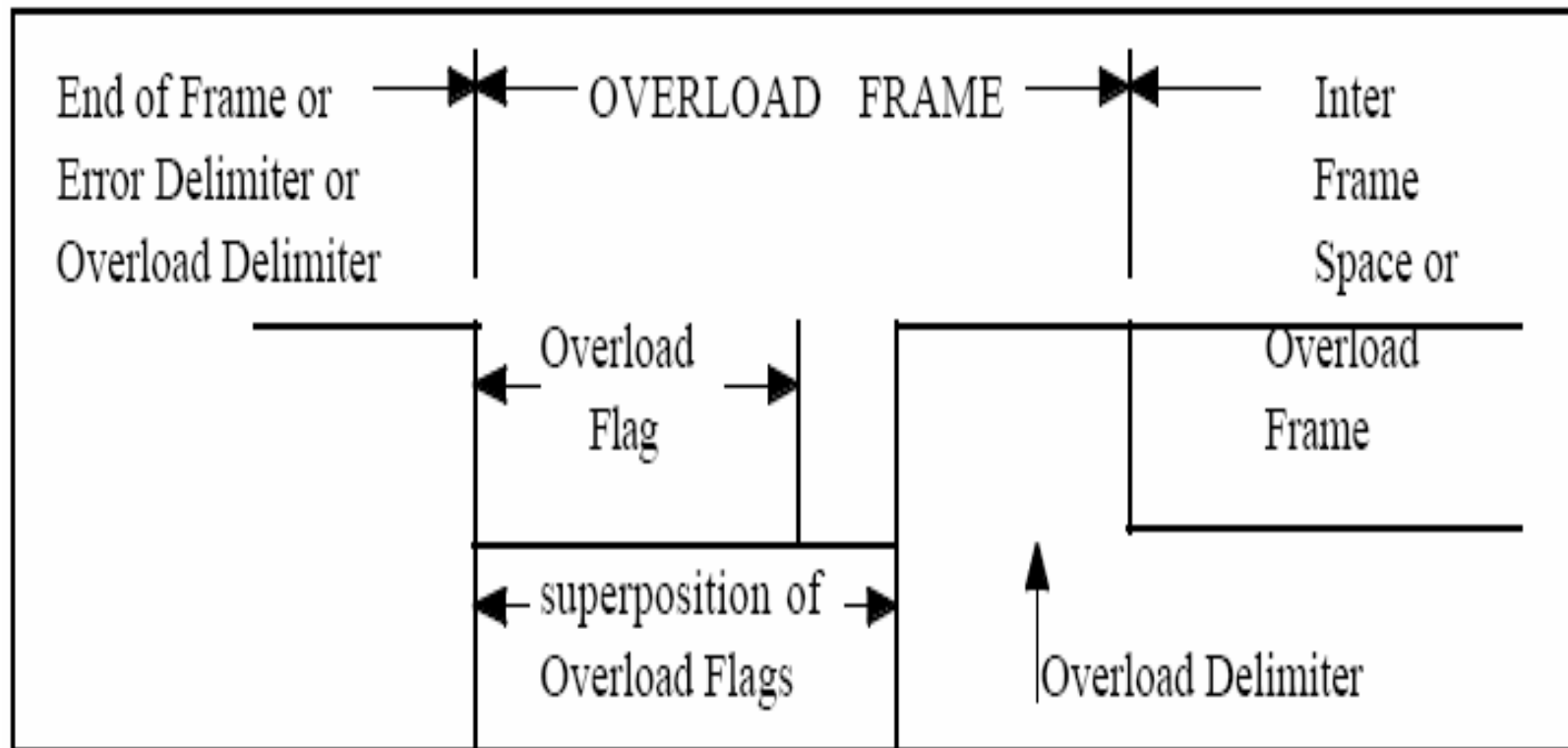
Có hai điều kiện quá tải:

- Điều kiện bên trong bộ nhận mà nó yêu cầu một thời gian trễ cho Data Frame và Remote Frame tiếp theo.
- Phát hiện ra một bit trội trong quá trình Intermission.

OVERLOAD FLAG bao gồm 6 bit trội.

OVERLOAD DELIMITER bao gồm 8 bit lặn

## CAN - Cấu trúc bản tin



## CAN - Cấu trúc bản tin

INTERFRAME SPACE bao gồm hai trường INTERMISSION và BUS IDLE

INTERMISSION bao gồm 3 bit lặn.

Các Data Frame và Remote Frame được cách nhau bởi INTERFRAME SPACE.

INTERMISSION các trạm không được truyền các Data hay Remote Frame mà chỉ có thể truyền điều khiển quá tải.

BUS IDLE trạm có thể truyền các bản tin.

# CAN

- Mã hoá thông tin: Các trường START OF FRAME, ARBITRATION FIELD, CONTROL FIELD, DATA FIELD and CRC SEQUENCE của bản tin được mã hoá bởi PP nhồi bit.
- Nếu bên phát phát hiện ra 5 bit liên nhau giống nhau trong chuỗi bit đã truyền đi thì nó sẽ tự động chèn một bit đảo vào.
- Các trường còn lại của DATA FRAME hay REMOTE FRAME (CRC DELIMITER, ACK FIELD and END OF FRAME) không bị chèn. ERROR FRAME và OVERLOAD FRAME duy trì định dạng cố định không bị chèn.
- các bit mã hoá NRZ.



## CAN - Kiểm soát lỗi.

Phát hiện lỗi: Có 5 kiểu lỗi khác nhau

- BIT ERROR
- STUFF ERROR
- CRC ERROR
- FORM ERROR
- ACKNOWLEDGEMENT ERROR

## CAN - Kiểm soát lỗi.

Báo lỗi: Một trạm khi phát hiện ra lỗi thì sẽ gửi đi một cờ lỗi. Các lỗi như BIT ERROR, STUFF ERROR, FORM ERROR ACKNOWLEDGEMENT ERROR phát hiện ra ở bit nào thì cờ lỗi sẽ truyền ở bit tiếp theo.

Nếu lỗi CRC được phát hiện thì cờ lỗi sẽ truyền sau ACK delimiter.

## CAN - Hạn chế lỗi

Hạn chế lỗi các nút CAN chia ra làm 3 trạng thái:

- error active
- error passive
- bus off

Mỗi một nút có hai bộ đếm:

**TRANSMIT ERROR COUNT**

**RECEIVE ERROR COUNT**

## CAN - Hạn chế lỗi

Các luật thay đổi giá trị bộ đếm:

1. Khi bộ nhận phát hiện ra 1 lỗi thì bộ đếm lỗi nhận tăng lên 1. Ngoại trừ trường hợp lỗi bit khi truyền cờ lỗi tích cực hay quá tải.
2. Khi bộ nhận phát hiện ra một bit trội là bit đầu tiên sau khi truyền đi cờ lỗi thì bộ đếm lỗi nhận tăng lên 8.
3. Khi bộ truyền phát hiện ra một lỗi thì nó sẽ truyền đi 1 cờ lỗi và bộ đếm lỗi truyền sẽ tăng lên 8.
4. Nếu bộ truyền phát hiện ra lỗi bit khi truyền cờ lỗi tích cực hay cờ quá tải thì bộ đếm lỗi truyền tăng lên 8.
5. Nếu bộ nhận phát hiện ra lỗi bit khi truyền cờ lỗi tích cực hay cờ quá tải thì bộ đếm lỗi nhận tăng lên 8.

## CAN - Hạn chế lỗi

6. Nếu bất kỳ nút nào chịu 7 bit trội liên nhau sau khi truyền đi cờ lỗi tích cực, bị động, cờ quá tải thì bộ đếm lỗi truyền và nhận tăng lên 8.
7. Sau khi truyền tốt một bản tin thì bộ đếm lỗi truyền giảm đi 1 ngoại trừ nó đã =0.
8. Sau khi nhận tốt một bản tin bộ đếm lỗi nhận:
  - giảm đi 1 nếu nó <127
  - =0 nếu =0.
  - 119-127 nếu nó >128.
9. Nút lỗi bị động nếu có BDLT hay BDLN >127.

## CAN - Hạn chế lỗi

- 10. Bus off nếu có BDLT > 256
- 11. Lỗi bị đông nếu BDLT và BDLN  $\leq 127$  .
- 12. Bus off  $\rightarrow$  lỗi tích cực với BDLT = BDLN = 0 sau 128 sự kiện 11 bit lặn liên nhau được ghi nhận trên Bus.

*CAN* - Hạn chế lỗi