CAN
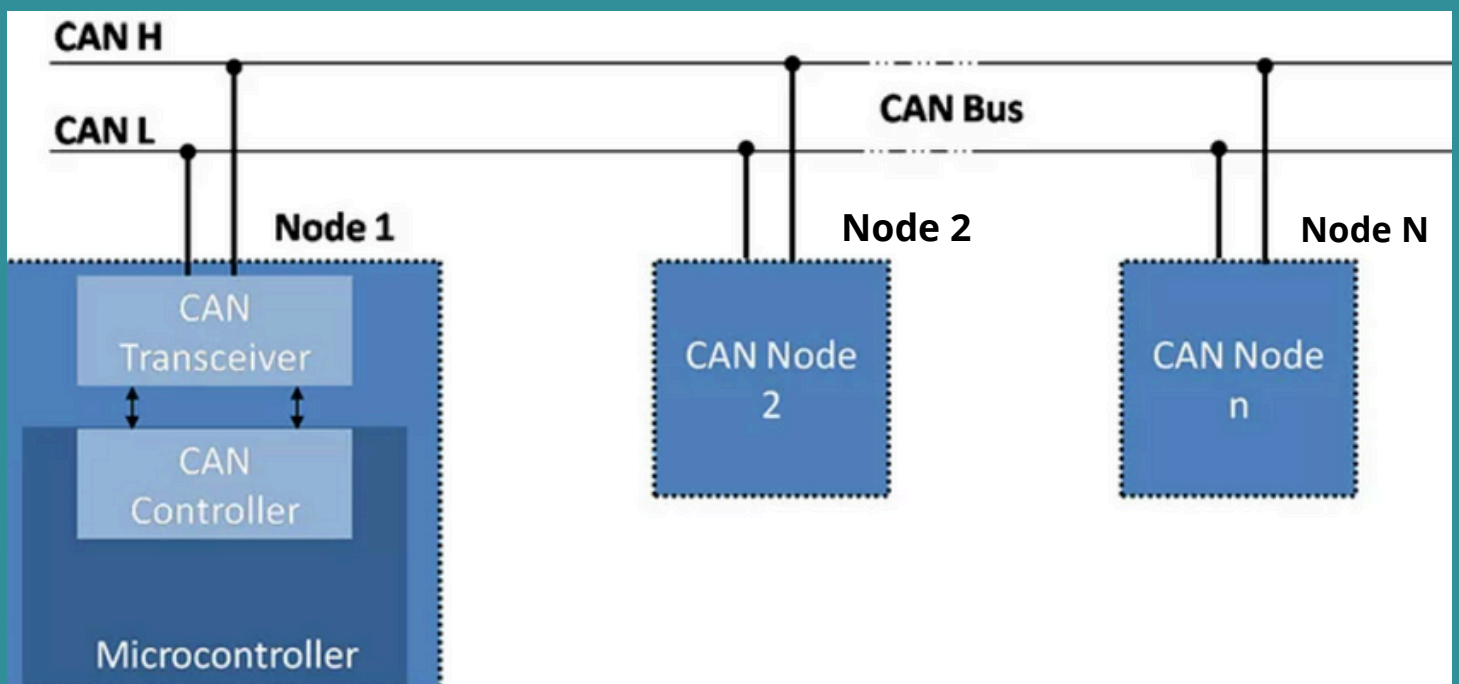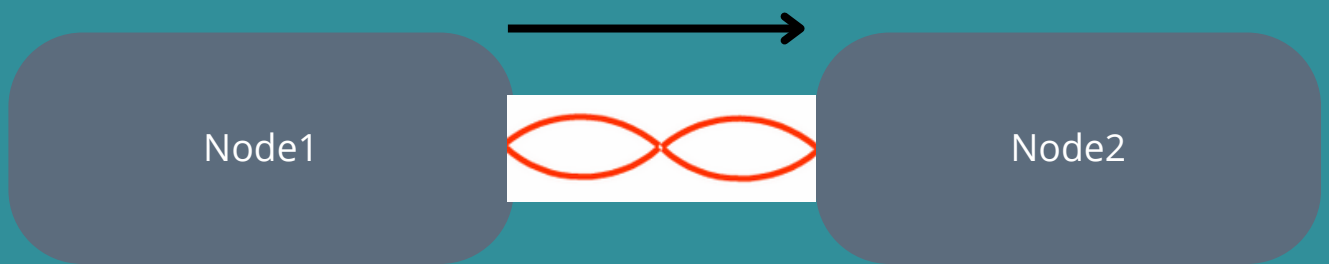## PROTOCAL

- **CAN** stands for Controller Area Network protocol.

- Designed to allow microcontrollers and devices to communicate with each other without a host computer.

- Asynchronous serial communication.

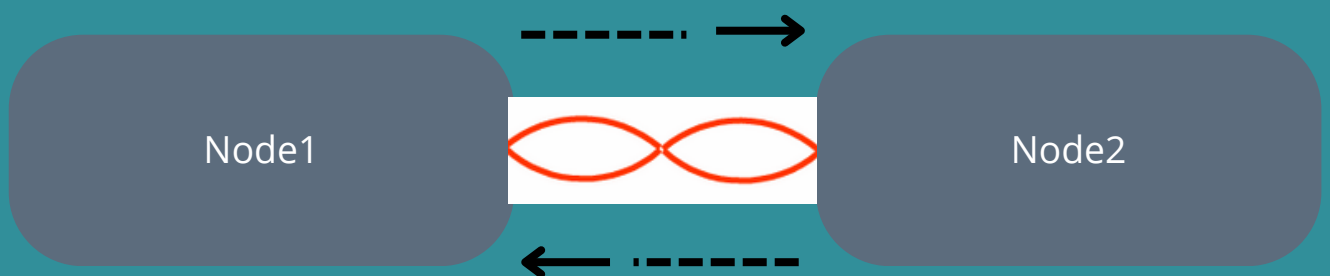- uses two wires for communication: CAN High (CAN_H) and CAN Low (CAN_L).

communication CAN can be:

- **Simplex** data Transmission in <u>one direction</u>.

- **Half-Duplex**: data transmission and reception.

About CAN

- One of the big advantage of **CAN** is an asynchronous serial communications protocol - <u>No Shared Clock</u>.

- Can is used for:

  Norme11898.2 High Speed (1Mbits/s)
  Norme 11898.3 Low Speed (125 Kbits/s)

- CAN enables a **MultiMaster** communication architecture, allowing multiple nodes on the network to independently initiate communication.

- All CAN bus nodes are wired to the bus via the principle of **<u>wired AND</u>**.

Transmission technology

- Logic states 0(dominant) and 1(recessive) are represented over there potential difference between CANH and CANL

- The **CAN** protocol defines two specific voltage levels to represent Recessive and Dominant bits.
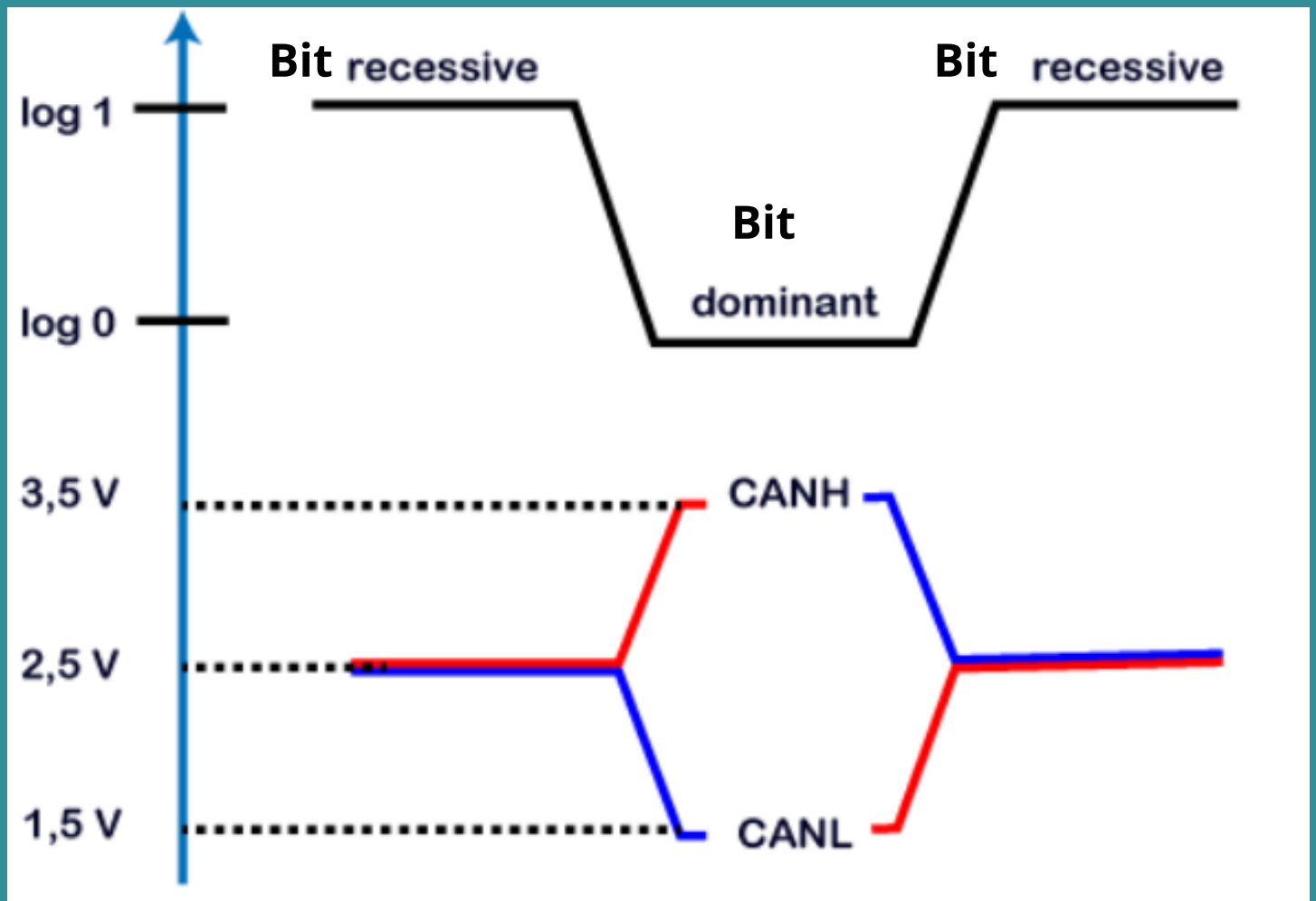
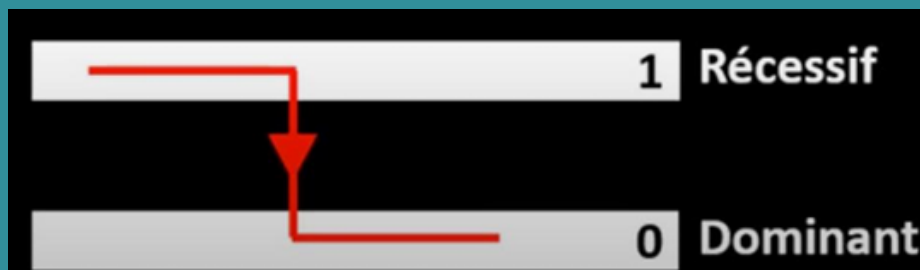The result of this example: 1 0 1

CAN frame format

The CAN standard have 4 type of frame:

- **Data frame**: 2 standards:

    CAN 2.0A <u>Standard</u> format: identifier field set to 11 bit.

    CAN 2.0B <u>Extended</u> Format: 29-bit identifier field.

- **Request frame**
- **Error frame**
- **Supplement frame**

The **CAN** frame contain:

- Start of frame(SOF): Transition from Recessive state (1) to Dominant state (0).

- End Of Frame: 7 bits Recessive.

- interface enter frame 3 bits Recessive.

Format standard CAN 2.0A
<u>Data frame</u>

S O F	11-bit Identifier	R T R	I D E	r0	DLC	0...8 Bytes Data	CRC	ACK	E O F	I F S

- <u>11-bit identifier</u>: The node identifier field.

- <u>RTR</u> (RemoteTransmission Request) bit:
    - "0" : data frame
    - "1" : request frame

- <u>IDE</u> (IDentifer Extended) bit :
    - "0" : Standard frame
    - "1" : Extended frame

- <u>DLC</u> (Data Length Code)4-bit Number of data bytes to transmit

- <u>0..8 Data</u> bytes: the data

- <u>CRC</u> (Cyclic Redundancy Check)16-bit

- <u>ACK</u>= Acknowledge + Delimitation bit

    Acknowledge bit:
    - "1": sent by the Transmitter
    - "0": The Receiver overwrites (good reception)

    Delimitation bit: 1-bit(recessive)

Format standard CAN 2.0A

Request frame
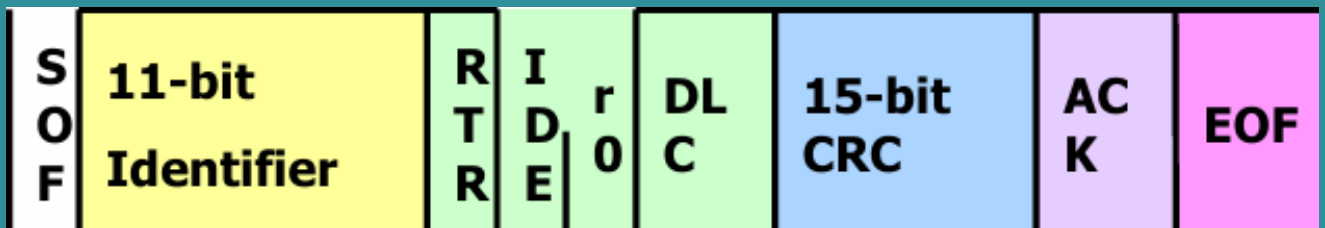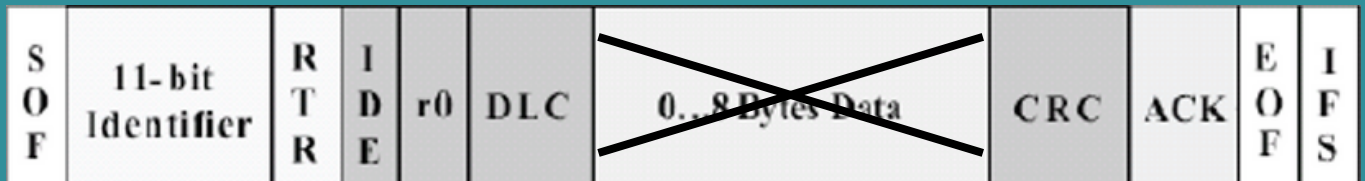
**RTR** takes the recessive state(1): meaning need of information

Error frame format

There are two types of error frames:

- Active error frames:

    1. Node A sends a data frame to Node C.

    2. Node B **detects** a type of error.

    3. Node B is in **active mode** and sending an active error frame.

- Passive error frames:

    1. Node A sends a data frame to Node C.
    2. while the node B is in **passive** mode but it **detected an erro**r.
    3. le Nœud B envoie une trame d'erreur passive.

Error frame format

- In a CAN protocol, when a node (like Node B) **detects** an error, it **sends** an **error** frame to **all nodes** on the network.

Error Flag   **Error Delimiter**

6 bits	8 bits
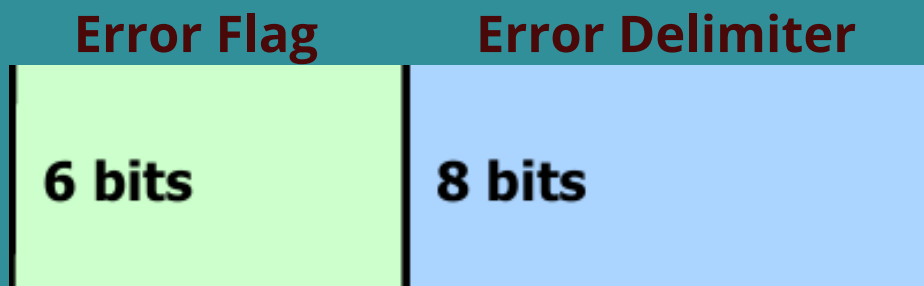
- **The types of errors:**
  - Bit Error
  - CRC Error
  - Stuff Error
  - Acknowledgment Error
  - Form Error

Priority and Arbitration

- The **priority** of a message is determined by the value of his **ID.**

- Any bus conflict is resolved by the "**wired AND**" mechanism.