

Control Area Network

- **Swarna Prabha Jena**

Asst. Prof. , Department of ECE

Contents

Sensors Used in Automotive
Types of Actuators Used in Automotive
Actuators in Automotive for the Vehicle Interior
Automotive Actuators present in Bodywork
Communication Protocols Used In Automotive
History of CAN Protocol
Controller Area Network
CAN in Automotive
CAN Versions
Difference Between High Speed CAN & Low Speed CAN
OSI Model
CAN Layered Architecture
CAN Bus Electrical Characteristics

Contents

Frame

Uses of Frame

CAN Frame Format

Types of CAN Frame

Data Frame

Remote Frame

Error Frame

Overload Frame

Interframe Space

Error Detection Mechanism

CAN Protocol Arbitration

BUS off State

Contents

4

Application of CAN Protocol
Advantages of CAN Protocol
Disadvantages of CAN Protocol

Sensors Used in Automotive

- **Mass Airflow Sensor**
- **Engine Speed sensor**
- **Oxygen Sensor**
- **Spark Knock Sensor**
- **Coolant Sensor**
- **Manifold Absolute Pressure (MAF) Sensor**
- **Fuel Temperature Sensor**
- **Voltage Sensor**
- **Camshaft Position Sensor**
- **Throttle Position Sensor**
- **Vehicle Speed Sensor**

Types of Actuators Used in Automotive

6

- Pneumatic
- Hydraulic
- Electromagnetic
- Gear Motors or Electric

Actuators in Automotive for the Vehicle Interior

7

- Steering Wheel
- Headrest
- Seats

Automotive Actuators Present in Bodywork

- Rear View Mirrors
- Sunroofs
- Headlights
- Lid of the gasoline deposit
- Engine Block
- Trunk
- Solenoids
- DC Motors
- Stepper Motors
- Piezo Actuators

Communication Protocols Used in Automotive

- The common automotive communication protocols are
 - CAN (Controller Area Network)
 - CAN FD (Controller Area Network Flexible Data-Rate)
 - LIN (Local Interconnect Network)
 - MOST (Media Oriented Systems Transport)
 - FlexRay
 - Bluetooth
 - Automotive Ethernet

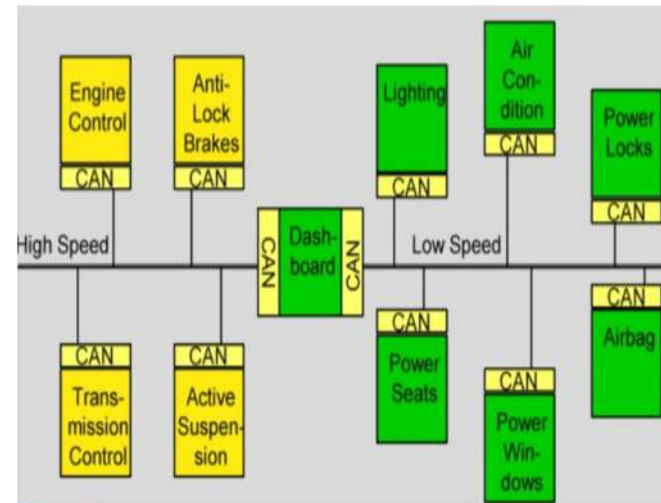
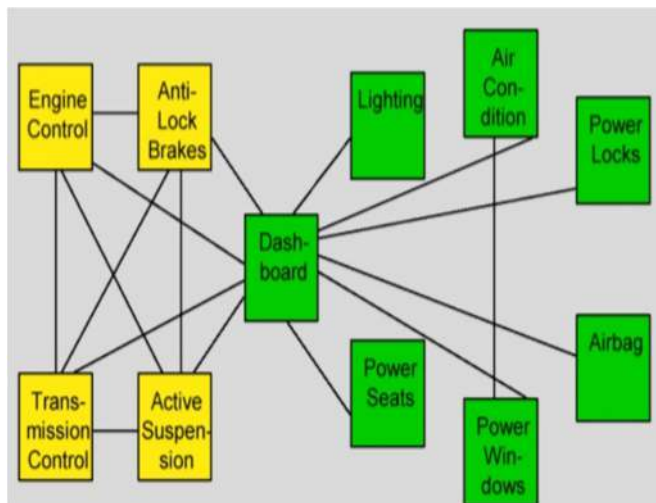
History of CAN Protocol

- It was developed by Robert Bosch in 1982.
- The protocol was officially released in 1986 at the Society of Automotive Engineers (SAE) conference in Detroit, Michigan.
- The first CAN controller chips, produced by Intel and Philips, came on the market in 1987.
- Released in 1991, the Mercedes-Benz W140 was the first production vehicle to feature a CAN-based multiplex wiring system.
- Bosch published several versions of the CAN specification and the latest is CAN 2.0 published in 1991.

Controller Area Network

- CAN stands for Controller Area Network.
- CAN is a serial half duplex asynchronous type of communication protocol.
- CAN bus is a standard which is designed to allow microcontrollers and other devices to communicate with each other without a host computer.
- CAN is a message based protocol means each message carry a message identifier based on which the priority of the message will be decided.
- It provides signaling rate from 125 kbps to 1 mbps.
- It provides 2048 message identifier for 11 bit and 536+ million message identifier for 29 bit.
- There is no need of node identification in the CAN network which makes it very simpler/flexible to insert or remove a node from the network.

CAN in Automotive



CAN Versions

- Latest Version of CAN is CAN 2.0 developed in the year 1991.
- This specification has two parts
 - CAN 2.0A
 - CAN 2.0B
- Part A is for the standard format with an 11 bit identifier.
- Part B is for the extended format with an 29 bit identifier.

Difference Between High Speed CAN & Low Speed CAN

High Speed CAN

- i. High Speed CAN works from 10 Kbits to 1 Mbits /sec.
- ii. High Speed CAN requires 120 ohm termination on the bus.
- iii. High Speed CAN won't work if There is problem in wiring.

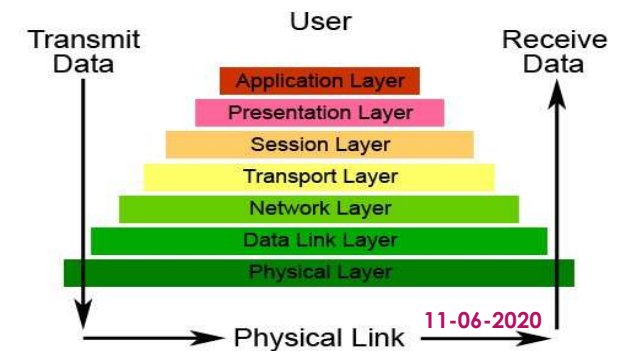
Low Speed CAN

- i. Low speed CAN restricts to 125 kbps.
- ii. Low speed CAN devices are Self terminated.
- iii. Low speed CAN work on asynchronous as it grounded at every node. That's why low speed CAN is called as Fault Tolerant CAN.

OSI Model

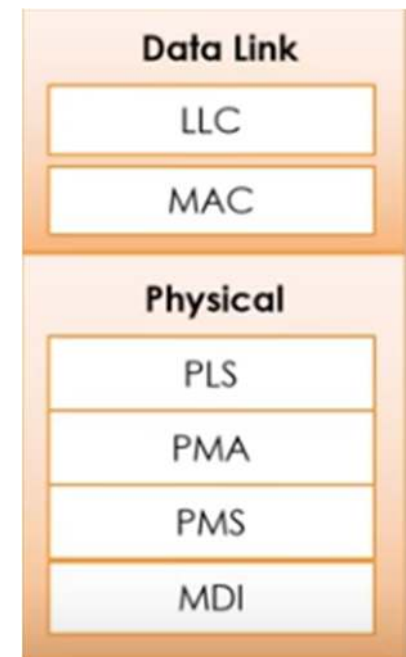
- OSI Model stands for Open System Interconnection Model.
- It is a conceptual model that partitions a system in 7 abstraction layers.
- It defines the communication function of each layer irrespective of the underlying internal structure & technology.
- According OSI reference model the CAN architecture defines two layers
 - Data Link Layer
 - Physical Layer

The Seven Layers of OSI



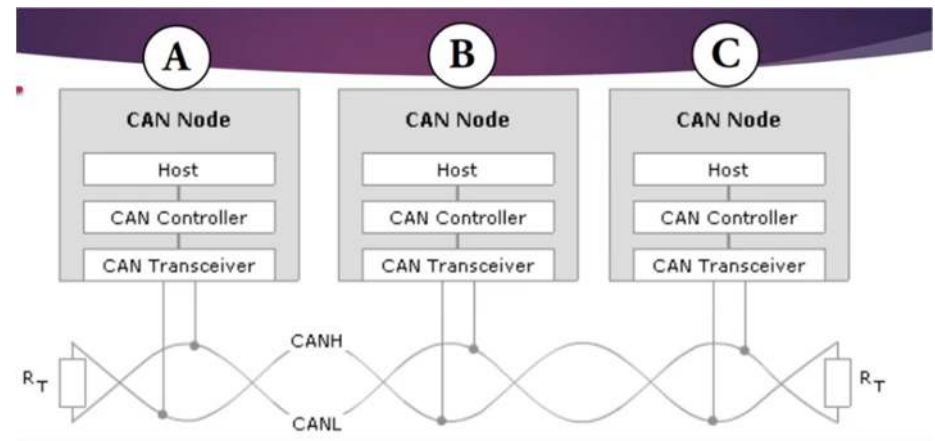
CAN Layered Architecture

- CAN architecture is divided into two parts.
 - Data Link Layer
 - Physical Layer
- Data Link layer is responsible for node to node data transfer.
- It defines the protocol to establish & terminate the connection
- & for also responsible for detecting & possibly correcting the
- error that may occur at the physical layer.
- Data link layer is divided into two sub layers
 - MAC(Medium Access Control)
 - LLC(Logical Link Control)
- Physical Layer is responsible for transmission of raw data.



CAN Bus Electrical Characteristics

- CAN network consists of a no of CAN nodes which are linked via a physical transmission medium.
- A modern CAN node consists of 3 sub elements.
- A host can be a microprocessor or a microcontroller which is actually running the core application to do some specific job.

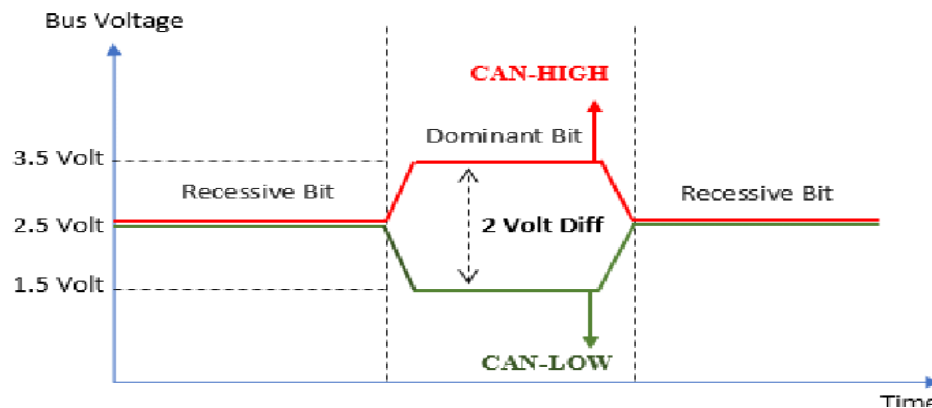


Contd...

- A host decides what the received message means & what message it want to send next.
- CAN controller deals with the communication function prescribed by the CAN protocol & it also triggers interrupt upon transmission reception of CAN messages.
- CAN transceiver which is responsible for transmission & reception of data on the CAN bus.
- The bus consists of two lines CAN High & CAN Low line which are also called CAN-H & CAN-L.
- The transmission happens with the help of differential voltage applied to this lines.

Contd...

- In CAN terminology a logic 1 is called recessive & logic 0 is called as dominant.
- When the CAN high line and CAN low line both are applied with 2.5 V then the actual differential voltage of the bus is 0 volt. A 0 volt on the CAN bus is read by can transceiver as a recessive or logic 1.
- When the CAN high line is pulled up to 3.5 volt & CAN low line is pulled down to 1.5 volt then the actual differential voltage of the bus is 2 volt which is treated as logic 0 or a dominant bit by CAN transceiver.



Frame

- Frame is a digital data transmission unit.
- A frame typically includes frame synchronization features i.e a sequence of bits that indicates to the receiver the beginning and end of the data.
- Frames help to determine how receivers interpret a stream of data from a source.

Uses of Frame

- Frame is used to send and receive data from one node to another node in the bus or in a network.
- It provides a way for a sender to transmit a set of bits that are meaningful to the receiver.

CAN Frame Format

- A CAN network can be configured to work with two different message (or "frame") formats:
 1. Standard or base frame format (described in CAN 2.0 A and CAN 2.0 B)
 2. Extended frame format (described only by CAN 2.0 B).
- The only difference between the two formats is that the "CAN base frame" supports a length of 11 bits for the identifier, and the "CAN extended frame" supports a length of 29 bits for the identifier, made up of the 11-bit identifier ("base identifier") and an 18-bit extension ("identifier extension").

Contd...

- Another difference between CAN base frame format and CAN extended frame format is made by using the IDE bit, which is transmitted as dominant in case of an 11-bit frame, and transmitted as recessive in case of a 29-bit frame.
- CAN controllers that support extended frame format messages are also able to send and receive messages in CAN base frame format.

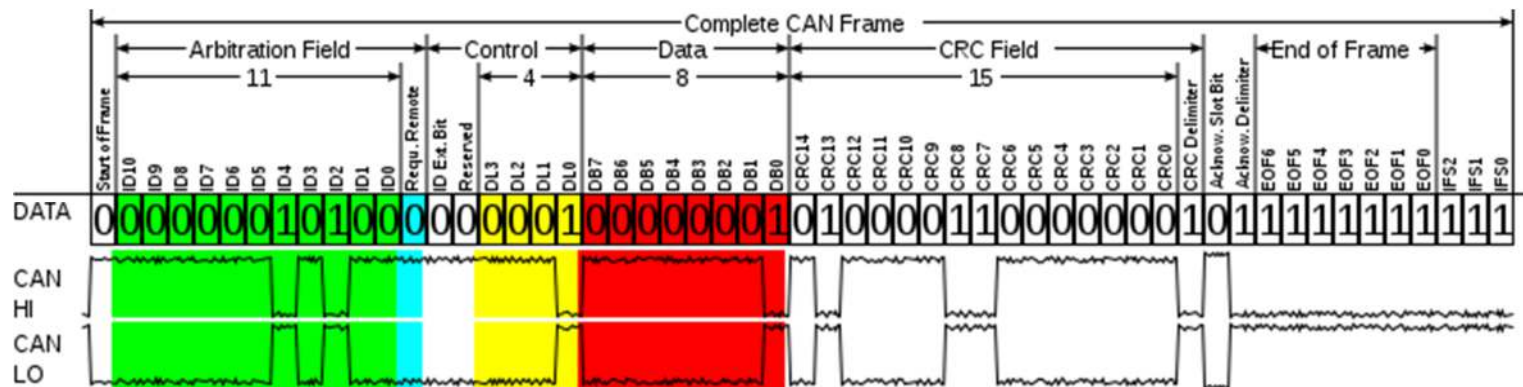
Types of CAN Frame

CAN has four frame types:

- Data frame: a frame containing node data for transmission.
- Remote frame: a frame requesting the transmission of a specific identifier.
- Error frame: a frame transmitted by any node detecting an error.
- Overload frame: a frame to inject a delay between data or remote frame.

Data Frame

- The data frame is the only frame for actual data transmission. There are two message formats:
- Base frame format: with 11 identifier bits
- Extended frame format: with 29 identifier bits



Contd...

- Can message start with a Start Of Frame(SOF) bit and ends with End Of Frame(EOF) bits.
- Start Of Frame(SOF) (1 bit) indicates the start of a new frame in the network and must be dominant.
- Identifier(11 bit) is unique in a network which is also represents the message priority(When multiple messages are transmitted at the same time).
- Remote Transmission Request(1 bit) must be dominant (0) for data frames and recessive (1) for remote request frames.
- Identifier extension bit (IDE) (1 bit) must be dominant (0) for base frame format with 11-bit identifiers and recessive(1) for extended frame format.

Contd...

- Reserved bit(2 bit) must be dominant (0), but accepted as either dominant or recessive .It is reserved for future extension.
- The number of bytes in the data field is indicated by the Data Length Code .This data length code is 4 bits wide and is transmitted within the control field.

DATA LENGTH CODE				DATA BYTE COUNT
DLC3	DLC2	DLC1	DLC0	
d	d	d	d	0
d	d	d	r	1
d	d	r	d	2
d	d	r	r	3
d	r	d	d	4
d	r	d	r	5
d	r	r	d	6
d	r	r	r	7
r	d	d	d	8
d = "dominant" r = "recessive"				

Contd...

- The data field consists of the data to be transferred within a Data frame .It can contain from 0 to 8 bytes ,each of which contain 8 bits which are transferred MSB first.
- A 15 bits Cyclic Redundancy Check or CRC field which is used to detect any data corruption during transmission.
- The CRC portion of the frame is obtained by selecting the input polynomial(to be divided) as the stream of bits from the start of frame(SOF) bit (included) to the Data field (if present) followed by 15 zeros.
- The polynomial is divided by the generator.

$$x^{15} + x^{14} + x^{10} + x^8 + x^7 + x^4 + x^3 + 1$$

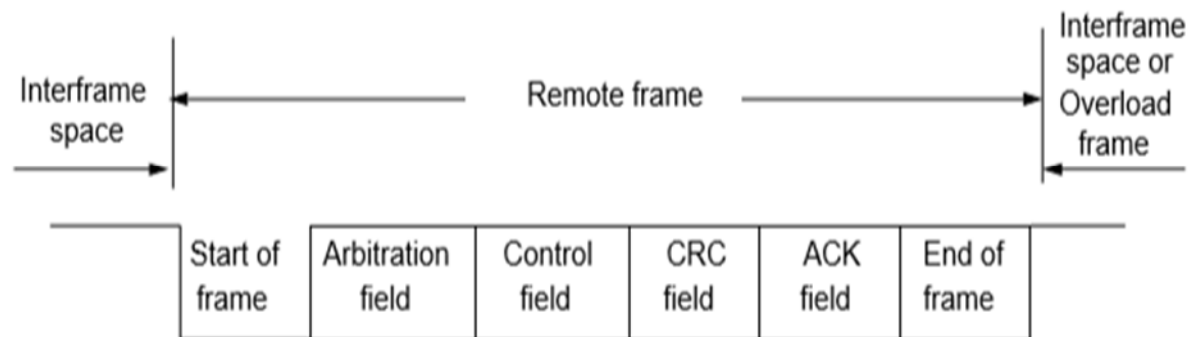
Contd...

- The remainder of the division is the CRC sequence portion of the frame.
- The CRC sequence is followed by the CRC Delimiter(1 bit) which consists of a single recessive bit.
- The ACK field is two bits long and contains the ACK slot(1 bit) and the ACK Delimiter(1 bit). In the ACK field the transmitting node sends two recessive bits.
- A receiver which has received a valid message correctly reports this to the TRANSMITTER by sending a dominant bit in the ACK slot.
- Frame ends with 7 consecutive recessive bits called as end of frame bits.

Remote Frame

- By sending a Remote Frame, a destination node can request the data from the source(it can be any node in the network).
- There are two differences between a Data Frame and a Remote Frame. 1.Firstly the RTR-bit is transmitted as a dominant bit in the Data Frame but recessive in remote frame.
2.secondly in the Remote Frame there is no Data Field. The DLC field indicates the data length of the requested message.
- In the event of a Data Frame and a Remote Frame with the same identifier being transmitted at the same time, the Data Frame wins arbitration due to the dominant RTR bit following the identifier.
- i.e. RTR = 0 ; DOMINANT in data frame
RTR = 1 ; RECESSIVE in remote frame

Contd...



Error Frame

- The error frame consists of two different fields:
- The first field is ERROR FLAGS (6 dominant/recessive bits).
- The following second field is the ERROR DELIMITER (8 recessive bits)

Active Error Flag

- six dominant bits – Transmitted by a node detecting an error on the network that is in error state "error active“.

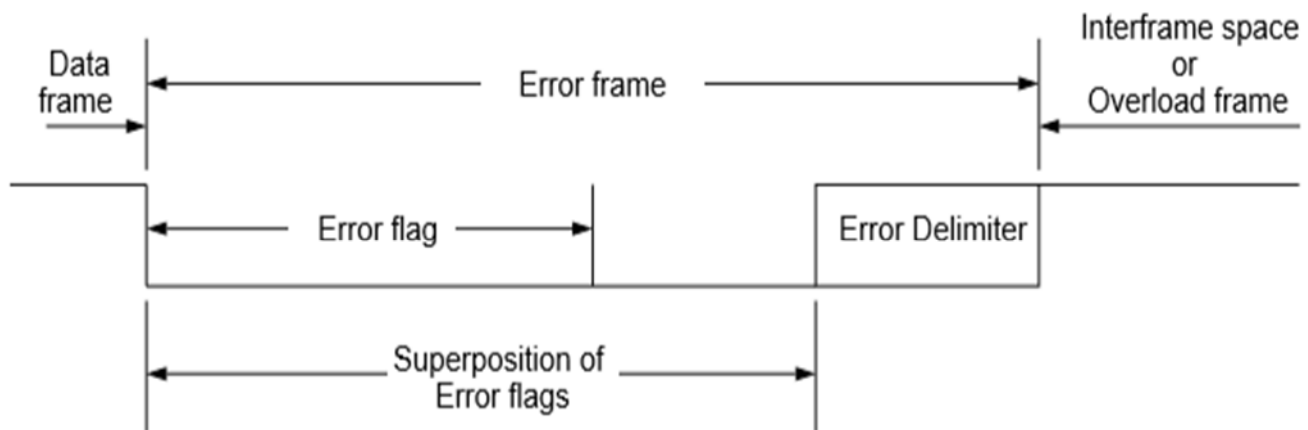
Passive Error Flag

- six recessive bits – Transmitted by a node detecting an active error frame on the network that is in error state "error passive“.

Contd...

- There are two error counters in CAN.
 1. Transmit error counter (TEC)
 2. Receive error counter (REC)
- A node starts out in Error Active mode.
- When any one of the two Error Counters raises above 127, the node will enter a state known as Error Passive .
- when the Transmit Error Counter raises above 255, the node will enter the Bus Off state.

Contd...



Contd...

- A node which is Bus Off will not transmit anything on the bus at all.
- The rules for increasing and decreasing the error counters are somewhat complex, but the principle is simple: transmit errors give 8 error points, and receive errors give 1 error point. Correctly transmitted and/or received messages causes the counter(s) to decrease.
- Example (slightly simplified): Let's assume that node A on a bus has a bad day.
- Whenever A tries to transmit a message, it fails (for whatever reason). Each time this happens, it increases its Transmit Error Counter by 8 and transmits an Active Error Flag. Then it will attempt to retransmit the message and the same thing happens.

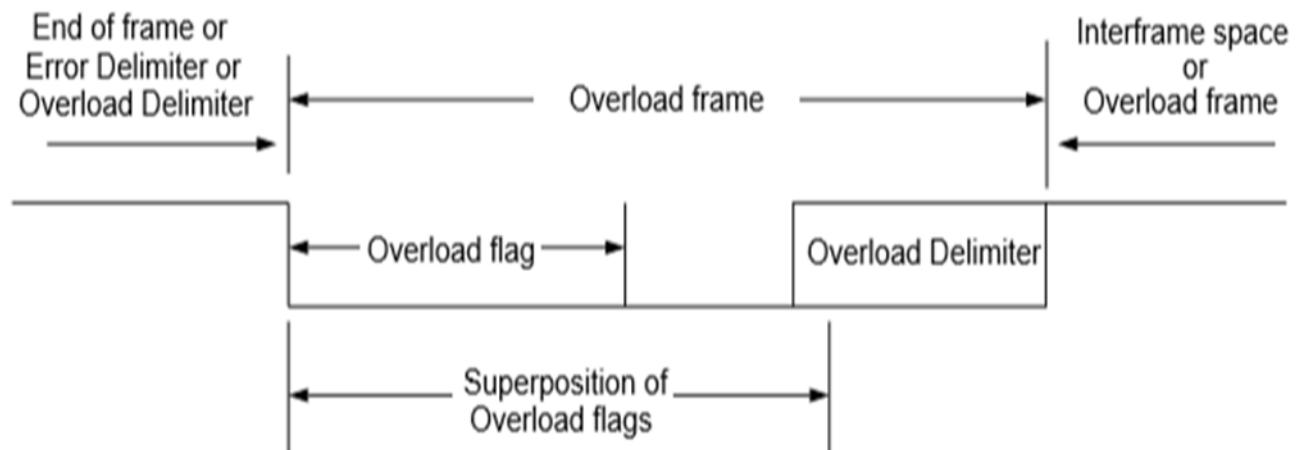
Contd...

- When the Transmit Error Counter raises above 127 (i.e. after 16 attempts), node A goes Error Passive.
- The difference is that it will now transmit Passive Error Flags on the bus. A Passive Error Flag comprises 6 recessive bits, and will not destroy other bus traffic so the other nodes will not hear A complaining about bus errors.
- However, A continues to increase its Transmit Error Counter. When it raises above 255, node A finally gives in and goes Bus Off.
- What does the other nodes think about node A?
 - For every active error flag that A transmitted, the other nodes will increase their Receive Error Counters by 1. By the time that A goes Bus Off, the other nodes will have a count in their Receive Error Counters that is well below the limit for Error Passive, i.e. 127.
- This count will decrease by one for every correctly received message. However, node A will stay bus off.

Overload frame

- Overload frame is used to inject a delay between data or remote frame.
- The overload frame contains the two bit fields Overload Flag and Overload Delimiter. There are two kinds of overload conditions that can lead to the transmission of an overload flag.
 1. The internal conditions of a receiver, which requires a delay of the next data frame or remote frame.
 2. Detection of a dominant bit during intermission.
- Overload Flag consists of six dominant bits. The overload flag's form destroys the fixed form of the intermission field.
- As a consequence, all other stations also detect an overload condition and on their part start transmission of an overload flag.
- Overload Delimiter consists of eight recessive bits.

Contd...



Inter Frame Space

- Data frames and remote frames are separated from preceding frames by a bit field called inter frame space.
- Inter frame space consists of at least three consecutive recessive bits.
- Following that, if a dominant bit is detected, it will be regarded as the "Start of frame" bit of the next frame.
- Overload frames and error frames are not preceded by an inter frame space and multiple overload frames are not separated by an inter frame space.

Error Detection Mechanisms

- The CAN protocol defines five different ways of detecting errors.
- Two of these works at the bit level, and the other three at the message level.
 - Bit Monitoring.
 - Bit Stuffing.
 - Frame Check.
 - Acknowledgement Check.
 - Cyclic Redundancy Check.

CAN Protocol Arbitration

- If two or more messages are ready to transmit on the bus at the same time by various nodes then which node will given the priority against the others to access the bus & continue its transmission.
- It is clear that the dominant state over writes the recessive state.
- When different CAN nodes sends dominant & recessive bus levels simultaneously, the bus at dominant and the recessive bus level occurs only if all node sends recessive state.
- This behavior is called as AND logic & physically implemented by open collector circuit.

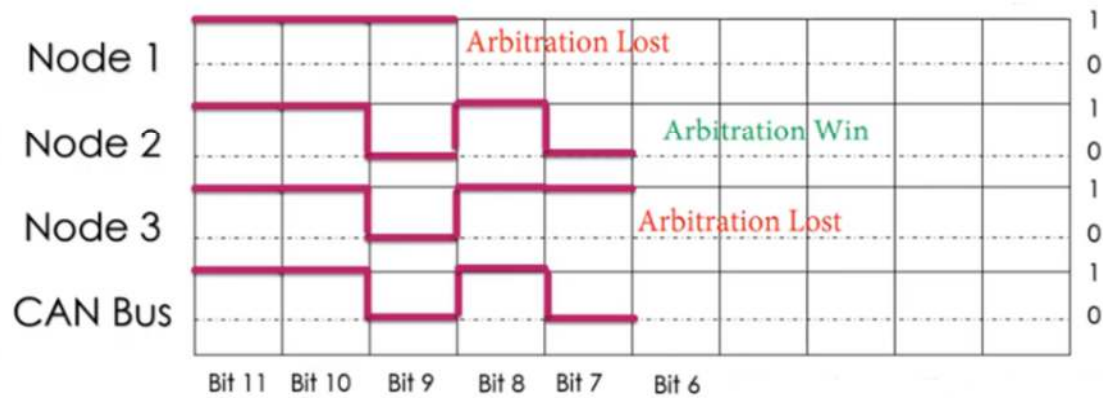
Wired-AND Logic

Sender 1	0	1	0	1	0	1	0	1
Sender 2	0	0	1	1	0	0	1	1
Sender 3	0	0	0	0	1	1	1	1
CAN Bus	0	0	0	0	0	0	0	1

Recessive

Contd...

- Example:-



CAN Node	Identifier (Hex)	Identifier (Binary)
1	0x7F3	11111110011
2	0x6B3	11010110011
3	0x6D9	11011011001

Contd...

- **Bit Monitoring**

Each transmitter on the CAN bus monitors (i.e. reads back) the transmitted signal level. If the bit level actually read differs from the one transmitted, a Bit Error is signaled.

- **Bit Stuffing**

When five consecutive bits of the same level have been transmitted by a node, it will add a sixth bit of the opposite level to the outgoing bit stream. The receivers will remove this extra bit. It gives the receivers an extra opportunity to detect errors: if more than five consecutive bits of the same level occurs on the bus, a Stuff Error is signaled.

Contd...

- **Frame check**

Some parts of the CAN message have a fixed format, i.e. the standard defines exactly what levels must occur and when. (Those parts are the SOF , CRC Delimiter, ACK Delimiter, End of Frame, and also the Intermission). If a CAN controller detects an invalid value in one of these fixed fields, a Frame Error is signaled.

- **Acknowledgement Check**

All nodes on the bus that correctly receives a message (regardless of their being “interested” of its contents or not) are expected to send a dominant level in the so-called Acknowledgement Slot in the message. The transmitter will transmit a recessive level here. If the transmitter can't detect a dominant level in the ACK slot, an Acknowledgement Error is signaled.

Contd...

- **Cyclic Redundancy Check**

Each message features a 15-bit Cyclic Redundancy Checksum (CRC), and any node that detects a different CRC in the message than what it has calculated itself will signal an CRC Error.

Bus Off State

- Whenever a CAN Transmitter error count reaches some threshold (say 255), a node will turn bus off and potentially reset itself. A good implementation will not continue resetting a node if the problem persists.
- Bus off means that a particular CAN node will get switched off from the CAN network. So there won't be any communication at all.
- If the engine controller goes offline, nearly every ECU in the vehicle will report "*Lost Communication with the Engine Controller.*" Typically, these type of CAN problems are identified by DTC's (Diagnostic Trouble Codes).

Contd...

- Depending on the severity of the issue, the vehicle might enter a "limp home" mode, or might be totally disabled. **Limp-home mode** is the condition when all the ECUs fail in the car network.
- A set of default parameters are initialized and your car can continue running only for some time before it is properly serviced by the OEM.
- A CAN bus node (ECU) automatically goes **bus on** after 128 x 11 bits, which is the equivalent for 128 messages.
- The 11 bits is the recessive time between messages so even in a 100% loaded bus, a bus off node will go bus on again.

Application of CAN Protocol

- Automobile(passenger Vehicle, Trucks, buses).
- Electronic equipment for aviation & navigation.
- Industrial automation & mechanical control.
- Elevators & escalators.
- Building Automation.
- Medical instrument.
- Marine, Military, Industrial equipment.

Advantages of CAN Protocol

- It allows 1Mbps data rate.
- It is used to reduce wiring in various automotive applications. Due to less complex interface, it is widely used across various industries.
- It saves overall cost and time due to less and simple wiring as well as use of flash programming.
- Supports auto retransmission of lost messages.
- It works in various electrical environments without any issues.
- The protocol supports different error detection capabilities such as bit error, ack error, form error, CRC error and stuff error.

Disadvantages of CAN Protocol

- Though maximum number of nodes are not specified for the network. It supports up to 64 nodes due to electrical loading.
- It supports maximum length of 40 meters.
- It is likely to have undesirable interactions between nodes.
- It incurs more expenditure for software development and maintenance.
- In order to reduce signal integrity issues such as reflections CAN bus should be properly terminated at both the ends with resistors.
- Node removal requires use of termination resistors of 120 Ohm value at appropriate places on the CAN bus.