Communication Protocols

CAN (Controller Area Network) protocol

CAN stands for **Controller Area Network** protocol. It is a protocol that was developed by **Robert Bosch** in around 1986. The CAN protocol is a standard designed to allow the microcontroller and other devices to communicate with each other without any host computer. The feature that makes the CAN protocol unique among other communication protocols is the broadcast type of bus. Here, broadcast means that the information is transmitted to all the nodes. The node can be a sensor, microcontroller, or a gateway that allows the computer to communicate over the network through the USB cable or ethernet port.

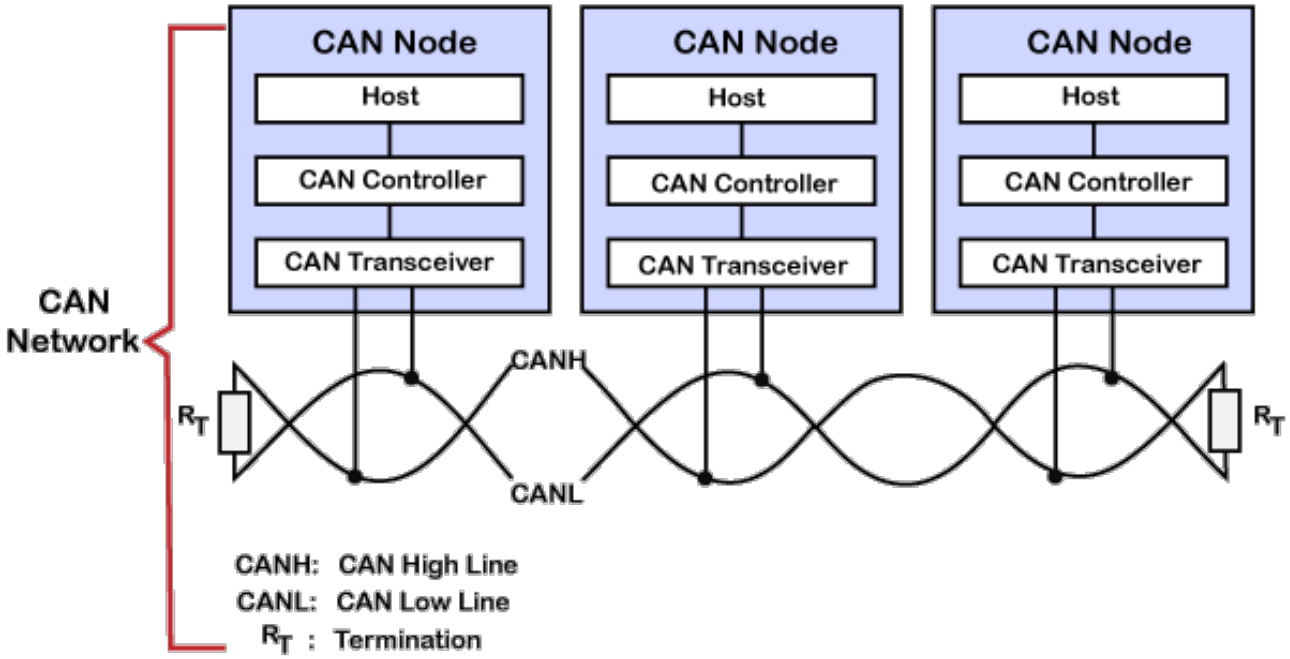
The CAN is a message-based protocol, which means that message carries the message identifier, and based on the identifier, priority is decided. There is no need for node identification in the CAN network, so it becomes very easy to insert or delete it from the network. It is a serial half-duplex and asynchronous type of communication protocol. The CAN is a two-wired communication protocol as the CAN network is connected through the two-wired bus. The wires are twisted pair having 120Ω characteristics impedance connected at each end. Initially, it was mainly designed for communication within the vehicles, but it is now used in many other contexts. Like UDS, and KWP 2000, CAN also be used for the on-board diagnostics.

Why CAN?

The need for a centralized standard communication protocol came because of the increase in the number of electronic devices. For example, there can be more than 7 TCU for various subsystems such as dashboard, transmission control, engine control unit, and many more in a modern vehicle. If all the nodes are connected one-to-one, then the speed of the communication would be very high, but the complexity and cost of the wires would be very high. In the above example, a single dashboard requires 8 connectors, so to overcome this issue, CAN was introduced as a

centralized solution that requires two wires, i.e., CAN high and CAN low. The solution of using CAN protocol is quite efficient due to its message prioritization, and flexible as a node can be inserted or removed without affecting the network.

Now we will see how data is transmitted through the CAN network.



A CAN network consists of multiple of CAN nodes. In the above case, we have considered three CAN nodes, and named them as node A, node B, and node C. CAN node consists of three elements which are given below:

- Host
  A host is a microcontroller or microprocessor which is running some application to do a specific job. A host decides what the received message means and what message it should send next.
- CAN Controller
  CAN controller deals with the communication functions described by the CAN protocol. It also triggers the transmission, or the reception of the CAN messages.
- CAN Transceiver
  CAN transceiver is responsible for the transmission or the reception of the data on the CAN bus. It converts the data signal into the stream

of data collected from the CAN bus that the CAN controller can understand.

Applications of CAN protocol

Initially, CAN protocol was designed to target the communication issue that occurs within the vehicles. But later on, due to the features it offers, it is used in various other fields. The following are the applications of CAN protocol:

- Automotive (passenger vehicles, trucks, buses)
- Electronic equipment for aviation and navigation
- Industrial automation and mechanical control
- Elevator and escalators
- Building automation
- Medical instruments and equipment
- Marine, medical, industrial, medical

I2C Communication Protocol

I2C stands for Inter-Integrated Circuit. It is a bus interface connection protocol incorporated into devices for serial communication. It was originally designed by Philips Semiconductor in 1982. Recently, it is a widely used protocol for short-distance communication. It is also known as Two Wired Interface(TWI).

Working of I2C Communication Protocol :
It uses only 2 bi-directional open-drain lines for data communication called SDA and SCL. Both these lines are pulled high.

Serial Data (SDA) – Transfer of data takes place through this pin.
Serial Clock (SCL) – It carries the clock signal.

I2C operates in 2 modes –

- Master mode
- Slave mode

Each data bit transferred on SDA line is synchronized by a high to the low pulse of each clock on the SCL line.

Differences:

CAN is a message based protocol, designed specifically for automotive applications for the communication between the main host controller and other devices (also now in industrial automation). It can be transferred to long distances. I2C is a short distance communication used between the ICs in a same circuit.

1. I2c is synchronous and CAN is asynshronous.
2. i2c needs slave address and CAN does not need slave address protocol.
3. i2c is Node oriented and CAN is message oriented.
4. i2c has SDA and SCL, CAN is differential bus..
5. i2c operates in 3 speeds 100kbps, 400kbps and 3.4mbps where as CAN operates at 250kbps upto 1mbps.