# CAN
Tutorial

CONTROLLER AREA NETWORK

CAN was designed by Bosch and is currently described by ISO 11898[1]. In terms of the Open Systems Interconnection model (OSI), CAN partially defines the services for layer 1 (physical) and layer 2 (data link). Other standards such as DeviceNet, Smart Distributed System, CAL, CAN Kingdom and CANopen (collectively called higher layer protocols) build upon the basic CAN specification and define additional services of the seven layer OSI model. Since all of these protocols utilize CAN integrated circuits, they all comply with the data link layer defined by CAN.

Figure 1.

The CAN Protocol Spec and the OSI m
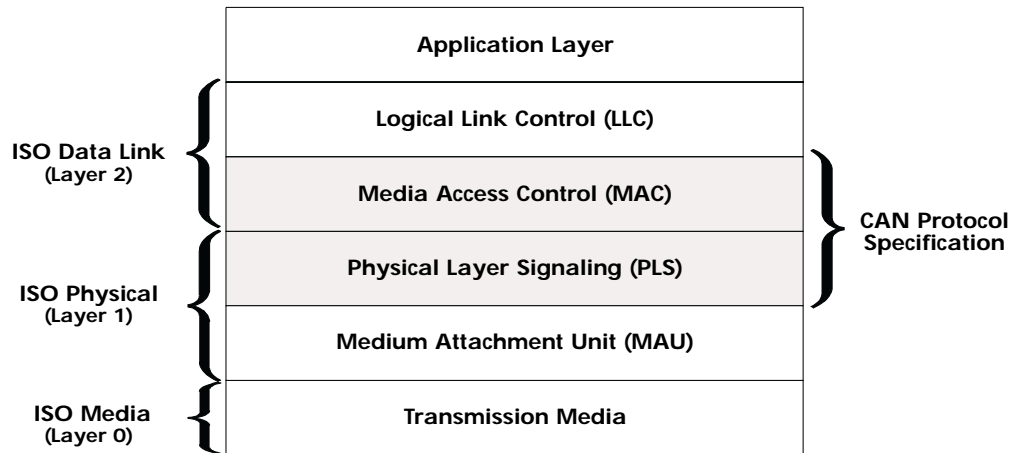
CAN specifies the medium access control (MAC) and physical layer signaling (PLS) as it applies to layers 1 and 2 of the OSI model. Medium access control is accomplished using a technique called non-destructive bit-wise arbitration. As stations apply their unique identifier to the network, they observe if their data are being faithfully produced. If it is not, the station assumes that a higher priority message is being sent and, therefore, halts transmission and reverts to receiving mode. The highest priority message gets through and the lower priority messages are resent at another time. The advantage of this approach is that collisions on the network do not destroy data and eventually all stations gain access to the network. The problem with this approach is that the arbitration is done on a bit by bit basis requiring all stations to hear one another within a bit-time (actually less than a bit-time). At a 500 kbps bit-rate, a bit-time is 2000 ns which does not allow much time for transceiver and cable delays. The result is that CAN networks are usually quite short and frequently less than 100 meters at higher speeds. To increase this distance either the data rate is decreased or additional equipment is required.

CAN DATA LINK LAYER

CAN transmissions operate using the producer/consumer model. When data are transmitted by a CAN device, no other devices are addressed. Instead, the content of the message is designated by an identifier field. This identifier field, which must be unique within the network, not only provides content but the priority of the message as well. All other CAN devices listen to the sender and accept only those messages of interest. This filtering of the data is accomplished using an acceptance filter which is an integral component of the CAN controller chip. Data which fail the acceptance criteria are rejected. Therefore, receiving devices consume only that data of interest from the producer.

A CAN frame consists mainly of an identifier field, a control field and a data field (Figure 2). The control field is six bits long, the data field is zero to eight bytes long and the identifier field is 11 bits long for standard frames (CAN specification 2.0A) or 29 bits long for extended frames (CAN specification 2.0B). Source and destination node addresses have no meaning using the CAN data link layer protocol.

Figure 2.
An 11-bit identifier is used in standard format.

Standard Format
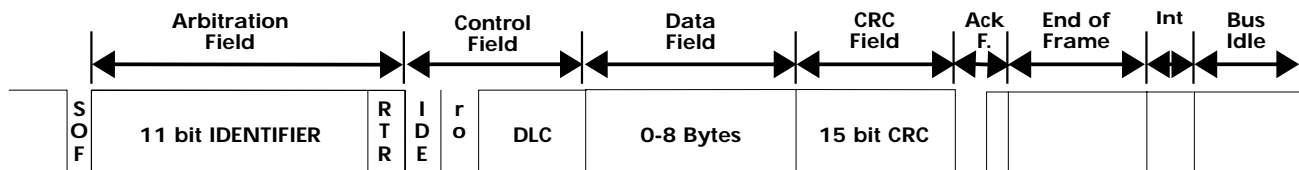
Bus arbitration is accomplished using a non-destructive bit-wise arbitration scheme. It is possible that more than one device may begin transmitting a message at the same time. Using a "wired AND" mechanism, a dominant state (logic 0) overwrites the recessive state (logic 1). As the various transmitters send their data out on the bus, they simultaneously listen for the faithful transmission of their data on a bit by bit basis until it is discovered that someone's dominant bit overwrote their recessive bit. This indicates that a device with a higher priority message, one with an identifier of lower binary value, is present and the loser of the arbitration immediately reverts to receiving mode and completes the reception of the message. With this approach no data are destroyed and, therefore, throughput is enhanced. The losers simply try again during their next opportunity. The problem with this scheme is that all devices must assert their data within the same bit-time and before the sampling point otherwise data will be falsely received or even destroyed. Therefore, a timing constraint has been introduced that impacts cabling distance.

PROPAGATION DELAY

In a Philips' application note[2], the author does an in-depth study on the maximum allowable propagation delay as a function of various controller chip parameters. The propagation delay (Figure 3) is due to the input/output delays of the CAN controller chip (tsd), transmission delay of the transceiver (ttx), reception delay of the transceiver (trx) and the signal delay due to the cable (tcbl). The total propagation delay (tp) experienced is basically the round trip delay from a CAN node located at the end of a cable segment communicating to the furthest node and is expressed as follows:

$$tp = 2(tsd+ttx+trx+tcbl)$$

All delays are constant except the cable delay (tcbl) which depends upon the length of the cable and the propagation delay factor of the cable (Pc). The author provides a chart of maximum allowable propagation delays (tpm) for various data rates and CAN chip timing parameters. The actual propagation delay must not exceed the maximum allowable propagation delay. By making the appropriate substitutions, we can determine the maximum allowable cable length (L).

$$L < \frac{\frac{1}{2}tpm\text{-}tsd\text{-}trx\text{-}ttx}{Pc}$$

Using appendix A.1 of the application note and the most favorable parameters for long distance, at 500 kbps, tpm equals 1626 ns. Assuming transceiver delays of 100 ns each, chip delay of 62.5 ns and a cable propagation factor of 5.5 ns/m, the maximum cable length is 100 meters which is the value used in the DeviceNet specification. Doing the same calculation at 250 kbps yields 248 meters and at 100 kbps, 680 meters. These values can be improved with better cable and faster transceivers.

Figure 3.
Use the longest path when calculating propagation delay.
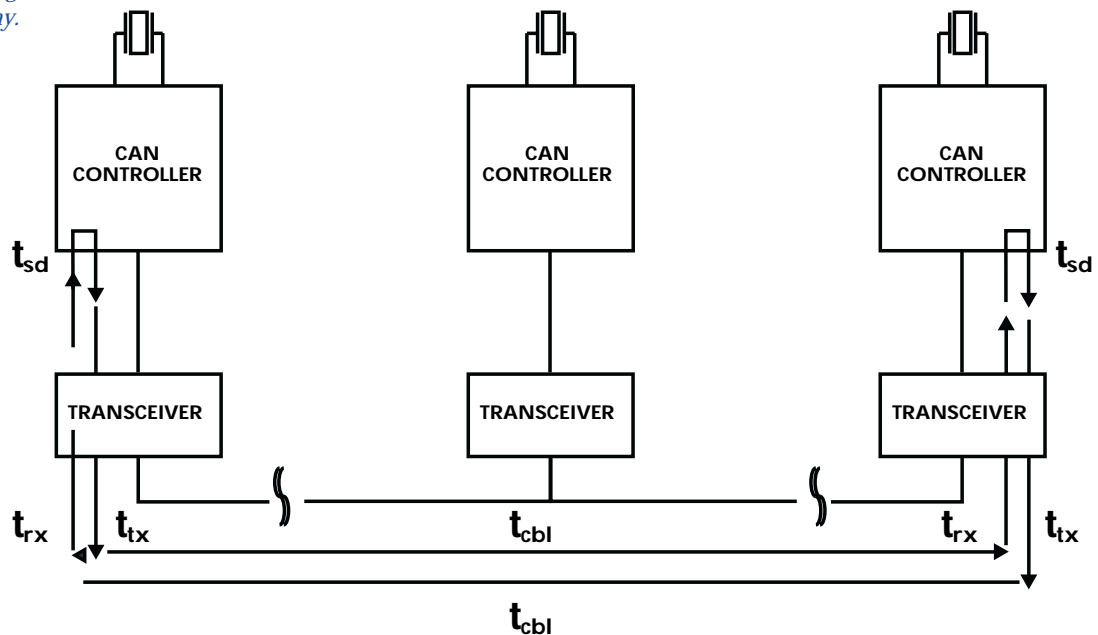
The point here is that CAN's bit-wise arbitration scheme inherently limits the maximum length of a CAN segment. Increasing the distance requires a reduction in data rate; however, there might be some benefit to incorporating repeaters or bridges.

REPEATERS

The usual approach to increasing network distance is to use repeaters. Repeaters provide signal boost to make up the loss of signal strength on a long segment. However, the problem with many CAN segments is not lack of signal strength but excessive signal latency. This latency is due to the propagation delay introduced by the transceivers and twisted-pair wiring. If this latency approaches one bit-time, the non-destructive bit-wise arbitration mechanism fails. Repeaters actually introduce more delay due to the additional electronics and are not effective in increasing the overall length of high speed CAN networks. Repeaters can be used to increase the effective length of drop cables from CAN trunk lines. Repeaters operate on the physical layer and are ignorant of the data link layer.

BRIDGES

Bridges are defined as devices that link two similar networks[3]. A local bridge stands by itself connecting adjacent wiring segments together as in the case of a repeater. Therefore, a local CAN bridge would have two CAN chips, one for one segment and one for the other. A microprocessor would pass messages between the two CAN chips. Using this approach, the effective length of the complete network is doubled while requiring only one bridge. Remote bridging interconnects two physically separated but similar networks together using a different interconnecting medium. Therefore, a pair of bridges are required to interconnect two networks the way two modems are used on leased phone lines. Sometimes bridges block network traffic by restricting data only to stations specified in the transmission that resided on the network controlled by the bridge. This blocking is difficult to implement in broadcast networks such as CAN and, therefore, is not recommended. Bridges operate at the data link layer and, therefore, are ignorant of the higher level protocols sent over CAN. As with the local bridge, two ports are required. However, instead of two CAN ports, one CAN port is replaced with a port compatible with the technology of the bridging connection. The technology chosen should be fast, deterministic, robust and capable of extending CAN networks without introducing excessive delay that would jeopardize the operation of the CAN system.

BIBLIOGRAPHY

1. ***Controller Area Network—A Serial Bus System—Not Just for Vehicles***, CAN in Automation (CiA)

2. Application Note, ***Bit Timing Parameters for CAN Networks***, Report No. KIE 07/91 ME, Philips Components, Buehring, Peter, 1991

3. ***PC Magazine Guide to Connectivity Second Edition***, Duerfler, Frank, Jr., Ziff-Davis Press, 1992

Tutorial

www.ccontrols.com

CONTEMPORARY CONTROLS