Introduction to the Controller Area Network (CAN)

Steve Corrigan   *Industrial Interface*

ABSTRACT

A controller area network (CAN) is ideally suited to the many high-level industrial protocols embracing CAN and ISO-11898:2003 as their physical layer. Its cost, performance, and upgradeability provide for tremendous flexibility in system design. This application report presents an introduction to the CAN fundamentals, operating principles, and the implementation of a basic CAN bus with TI's CAN transceivers and DSPs. The electrical layer requirements of a CAN bus are discussed along with the importance of the different features of a TI CAN transceiver.

Contents

List of Figures

List of Tables

Trademarks

CANopen is a trademark of CAN in Automation.
DeviceNet is a trademark of Open DeviceNet Vendor Association, Inc.

1   Introduction

The CAN bus was developed by BOSCH [1] as a multi-master, message broadcast system that specifies a maximum signaling rate of 1 megabit per second (bps). Unlike a traditional network such as USB or Ethernet, CAN does not send large blocks of data point-to-point from node A to node B under the supervision of a central bus master. In a CAN network, many short messages like temperature or RPM are broadcast to the entire network, which provides for data consistency in every node of the system.

Once CAN basics such as message format, message identifiers, and bit-wise arbitration -- a major benefit of the CAN signaling scheme are explained, a CAN bus implementation is examined, typical waveforms presented, and transceiver features examined.

2   The CAN Standard

CAN is an International Standardization Organization (ISO) defined serial communications bus originally developed for the automotive industry to replace the complex wiring harness with a two-wire bus. The specification calls for high immunity to electrical interference and the ability to self-diagnose and repair data errors. These features have led to CAN's popularity in a variety of industries including building automation, medical, and manufacturing.

The CAN communications protocol, ISO-11898: 2003, describes how information is passed between devices on a network and conforms to the Open Systems Interconnection (OSI) model that is defined in terms of layers. Actual communication between devices connected by the physical medium is defined by the physical layer of the model. The ISO 11898 architecture defines the lowest two layers of the seven layer OSI/ISO model as the data-link layer and physical layer in Figure 1.
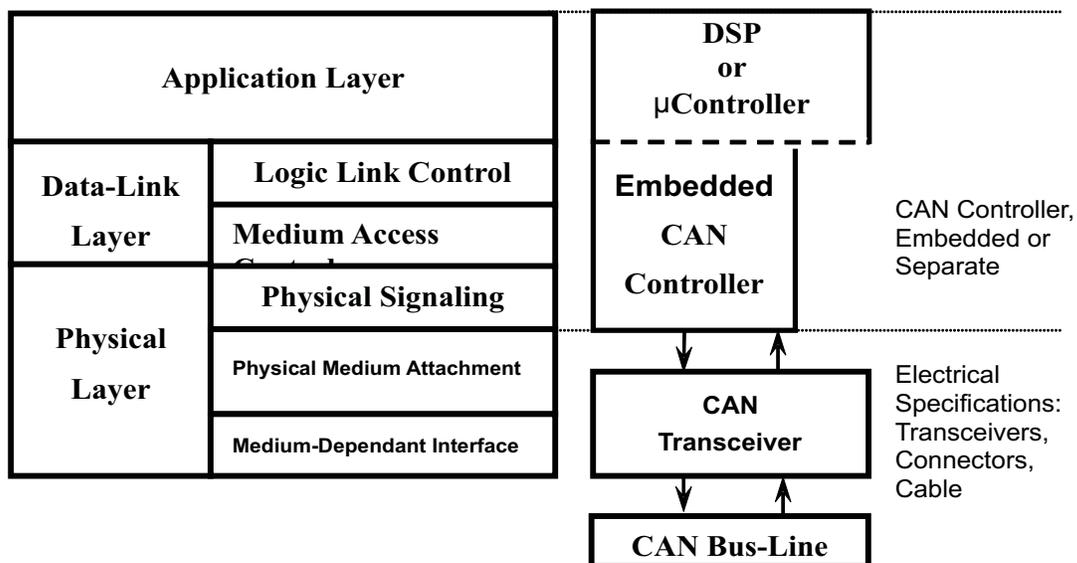

Figure 1. The Layered ISO 11898 Standard Architecture

In Figure 1, the application layer establishes the communication link to an upper-level application specific protocol such as the vendor-independent CANopen™ protocol. This protocol is supported by the international users and manufacturers group, CAN in Automation (CiA). Additional CAN information is located at the CiA Web site, can-cia.de. Many protocols are dedicated to particular applications like industrial automation, diesel engines, or aviation. Other examples of industry-standard, CAN-based protocols are KVASER's CAN Kingdom and Rockwell Automation's DeviceNet™.

---

[1]   Robert Bosch GmbH, www.bosch.com

3 Standard CAN or Extended CAN

The CAN communication protocol is a carrier-sense, multiple-access protocol with collision detection and arbitration on message priority (CSMA/CD+AMP). CSMA means that each node on a bus must wait for a prescribed period of inactivity before attempting to send a message. CD+AMP means that collisions are resolved through a bit-wise arbitration, based on a preprogrammed priority of each message in the identifier field of a message. The higher priority identifier always wins bus access. That is, the last logic-high in the identifier keeps on transmitting because it is the highest priority. Since every node on a bus takes part in writing every bit "as it is being written," an arbitrating node knows if it placed the logic-high bit on the bus.

The ISO-11898:2003 Standard, with the standard 11-bit identifier, provides for signaling rates from 125 kbps to 1 Mbps. The standard was later amended with the "extended" 29-bit identifier. The standard 11-bit identifier field in Figure 2 provides for 2^{11}, or 2048 different message identifiers, whereas the extended 29-bit identifier in Figure 3 provides for 2^{29}, or 537 million identifiers.

3.1 The Bit Fields of Standard CAN and Extended CAN

3.1.1 Standard CAN

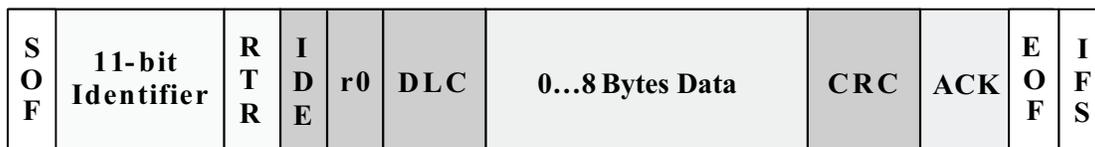| S O F	11-bit Identifier	R T R	I D E	r0	DLC	0…8 Bytes Data	CRC	ACK	E O F	I F S

Figure 2. Standard CAN: 11-Bit Identifier

The meaning of the bit fields of Figure 2 are:

- SOF–The single dominant start of frame (SOF) bit marks the start of a message, and is used to synchronize the nodes on a bus after being idle.
- Identifier-The Standard CAN 11-bit identifier establishes the priority of the message. The lower the binary value, the higher its priority.
- RTR–The single remote transmission request (RTR) bit is dominant when information is required from another node. All nodes receive the request, but the identifier determines the specified node. The responding data is also received by all nodes and used by any node interested. In this way, all data being used in a system is uniform.
- IDE–A dominant single identifier extension (IDE) bit means that a standard CAN identifier with no extension is being transmitted.
- r0–Reserved bit (for possible use by future standard amendment).
- DLC–The 4-bit data length code (DLC) contains the number of bytes of data being transmitted.
- Data–Up to 64 bits of application data may be transmitted.
- CRC–The 16-bit (15 bits plus delimiter) cyclic redundancy check (CRC) contains the checksum (number of bits transmitted) of the preceding application data for error detection.
- ACK–Every node receiving an accurate message overwrites this recessive bit in the original message with a dominate bit, indicating an error-free message has been sent. Should a receiving node detect an error and leave this bit recessive, it discards the message and the sending node repeats the message after rearbitration. In this way, each node acknowledges (ACK) the integrity of its data. ACK is 2 bits, one is the acknowledgment bit and the second is a delimiter.
- EOF–This end-of-frame (EOF), 7-bit field marks the end of a CAN frame (message) and disables bit-stuffing, indicating a stuffing error when dominant. When 5 bits of the same logic level occur in succession during normal operation, a bit of the opposite logic level is *stuffed* into the data.
- IFS–This 7-bit interframe space (IFS) contains the time required by the controller to move a correctly received frame to its proper position in a message buffer area.

3.1.2 Extended CAN

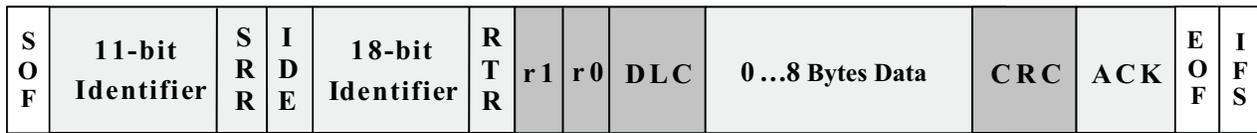| S O F	11-bit Identifier	S R R	I D E	18-bit Identifier	R T R	r1	r0	DLC	0…8 Bytes Data	CRC	ACK	E O F	I F S

Figure 3. Extended CAN: 29-Bit Identifier

As shown in Figure 3, the Extended CAN message is the same as the Standard message with the addition of:

- SRR–The substitute remote request (SRR) bit replaces the RTR bit in the standard message location as a placeholder in the extended format.
- IDE–A recessive bit in the identifier extension (IDE) indicates that more identifier bits follow. The 18-bit extension follows IDE.
- r1–Following the RTR and r0 bits, an additional reserve bit has been included ahead of the DLC bit.

4 A CAN Message

4.1 Arbitration

A fundamental CAN characteristic shown in Figure 4 is the opposite logic state between the bus, and the driver input and receiver output. Normally, a logic-high is associated with a one, and a logic-low is associated with a zero - but not so on a CAN bus. This is why TI CAN transceivers have the driver input and receiver output pins passively pulled high internally, so that in the absence of any input, the device automatically defaults to a recessive bus state on all input and output pins.
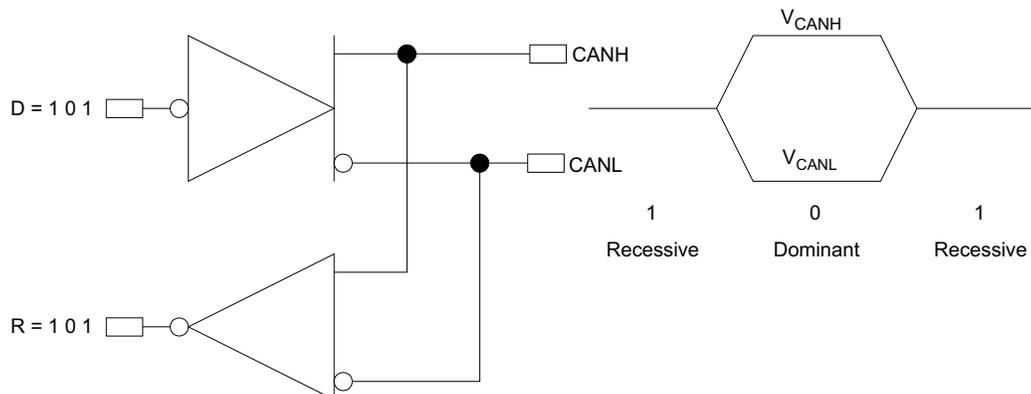

Figure 4. The Inverted Logic of a CAN Bus

Bus access is event-driven and takes place randomly. If two nodes try to occupy the bus simultaneously, access is implemented with a nondestructive, bit-wise arbitration. Nondestructive means that the node winning arbitration just continues on with the message, without the message being destroyed or corrupted by another node.

The allocation of priority to messages in the identifier is a feature of CAN that makes it particularly attractive for use within a real-time control environment. The lower the binary message identifier number, the higher its priority. An identifier consisting entirely of zeros is the highest priority message on a network because it holds the bus dominant the longest. Therefore, if two nodes begin to transmit simultaneously, the node that sends a last identifier bit as a zero (dominant) while the other nodes send a one (recessive) retains control of the CAN bus and goes on to complete its message. A dominant bit always overwrites a recessive bit on a CAN bus.

Note that a transmitting node constantly monitors each bit of its own transmission. This is the reason for the transceiver configuration of Figure 4 in which the CANH and CANL output pins of the driver are internally tied to the receiver's input. The propagation delay of a signal in the internal loop from the driver input to the receiver output is typically used as a qualitative measure of a CAN transceiver. This propagation delay is referred to as the loop time (tLOOP in a TI data sheet), but takes on varied nomenclature from vendor to vendor.

Figure 5 displays the CAN arbitration process that is handled automatically by a CAN controller. Because each node continuously monitors its own transmissions, as node B's recessive bit is overwritten by node C's higher priority dominant bit, B detects that the bus state does not match the bit that it transmitted. Therefore, node B halts transmission while node C continues on with its message. Another attempt to transmit the message is made by node B once the bus is released by node C. This functionality is part of the ISO 11898 physical signaling layer, which means that it is contained entirely within the CAN controller and is completely transparent to a CAN user.
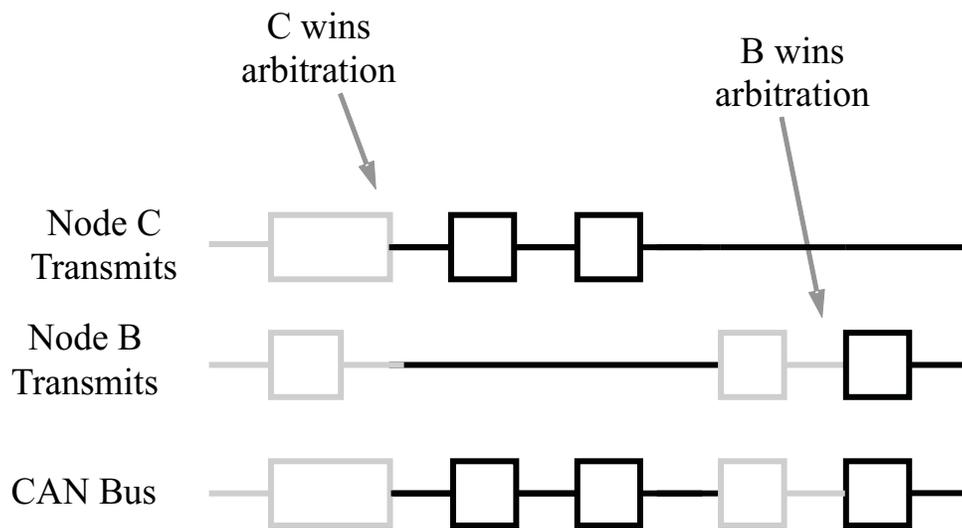

Figure 5. Arbitration on a CAN Bus

The allocation of message priority is up to a system designer, but industry groups mutually agree on the significance of certain messages. For example, a manufacturer of motor drives may specify that message 0010 is a winding current feedback signal from a motor on a CAN network and that 0011 is the tachometer speed. Because 0010 has the lowest binary identifier, messages relating to current values always have a higher priority on the bus than those concerned with tachometer readings.

In the case of DeviceNet™, devices from many manufacturers such as proximity switches and temperature sensors can be incorporated into the same system. Because the messages generated by DeviceNet sensors have been predefined by their professional association, the Open DeviceNet Vendors Association (ODVA), a certain message always relates to the specific type of sensor such as temperature, regardless of the actual manufacturer.

4.2   Message Types

The four different message types, or frames (see Figure 2 and Figure 3), that can be transmitted on a CAN bus are the data frame, the remote frame, the error frame, and the overload frame.

4.2.1   The Data Frame

The data frame is the most common message type, and comprises the Arbitration Field, the Data Field, the CRC Field, and the Acknowledgment Field. The Arbitration Field contains an 11-bit identifier in Figure 2 and the RTR bit, which is dominant for data frames. In Figure 3, it contains the 29-bit identifier and the RTR bit. Next is the Data Field which contains zero to eight bytes of data, and the CRC Field which contains the 16-bit checksum used for error detection. Last is the Acknowledgment Field.

4.2.2   The Remote Frame

The intended purpose of the remote frame is to solicit the transmission of data from another node. The remote frame is similar to the data frame, with two important differences. First, this type of message is explicitly marked as a remote frame by a recessive RTR bit in the arbitration field, and secondly, there is no data.

4.2.3   The Error Frame

The error frame is a special message that violates the formatting rules of a CAN message. It is transmitted when a node detects an error in a message, and causes all other nodes in the network to send an error frame as well. The original transmitter then automatically retransmits the message. An elaborate system of error counters in the CAN controller ensures that a node cannot tie up a bus by repeatedly transmitting error frames.

4.2.4   The Overload Frame

The overload frame is mentioned for completeness. It is similar to the error frame with regard to the format, and it is transmitted by a node that becomes too busy. It is primarily used to provide for an extra delay between messages.

4.3   A Valid Frame

A message is considered to be error free when the last bit of the ending EOF field of a message is received in the error-free recessive state. A dominant bit in the EOF field causes the transmitter to repeat a transmission.

4.4   Error Checking and Fault Confinement

The robustness of CAN may be attributed in part to its abundant error-checking procedures. The CAN protocol incorporates five methods of error checking: three at the message level and two at the bit level. If a message fails any one of these error detection methods, it is not accepted and an error frame is generated from the receiving node. This forces the transmitting node to resend the message until it is received correctly. However, if a faulty node hangs up a bus by continuously repeating an error, its transmit capability is removed by its controller after an error limit is reached.

Error checking at the message level is enforced by the CRC and the ACK slots displayed in Figure 2 and Figure 3. The 16-bit CRC contains the checksum of the preceding application data for error detection with a 15-bit checksum and 1-bit delimiter. The ACK field is two bits long and consists of the acknowledge bit and an acknowledge delimiter bit.

Also at the message level is a form check. This check looks for fields in the message which must always be recessive bits. If a dominant bit is detected, an error is generated. The bits checked are the SOF, EOF, ACK delimiter, and the CRC delimiter bits

At the bit level, each bit transmitted is monitored by the transmitter of the message. If a data bit (not arbitration bit) is written onto the bus and its opposite is read, an error is generated. The only exceptions to this are with the message identifier field which is used for arbitration, and the acknowledge slot which requires a recessive bit to be overwritten by a dominant bit.

The final method of error detection is with the bit-stuffing rule where after five consecutive bits of the same logic level, if the next bit is not a complement, an error is generated. Stuffing ensures that rising edges are available for on-going synchronization of the network. Stuffing also ensures that a stream of bits are not mistaken for an error frame, or the seven-bit interframe space that signifies the end of a message. Stuffed bits are removed by a receiving node's controller before the data is forwarded to the application.

With this logic, an active error frame consists of six dominant bits—violating the bit stuffing rule. This is interpreted as an error by all of the CAN nodes which then generate their own error frame. This means that an error frame can be from the original six bits to twelve bits long with all the replies. This error frame is then followed by a delimiter field of eight recessive bits and a bus idle period before the corrupted message is retransmitted. It is important to note that the retransmitted message still has to contend for arbitration on the bus.

5 The CAN Bus

The data link and physical signaling layers of Figure 1, which are normally transparent to a system operator, are included in any controller that implements the CAN protocol, such as TI's TMS320LF2812 3.3-V DSP with integrated CAN controller. Connection to the physical medium is then implemented through a line transceiver such as TI's SN65HVD230 3.3-V CAN transceiver to form a system node as shown in Figure 6.
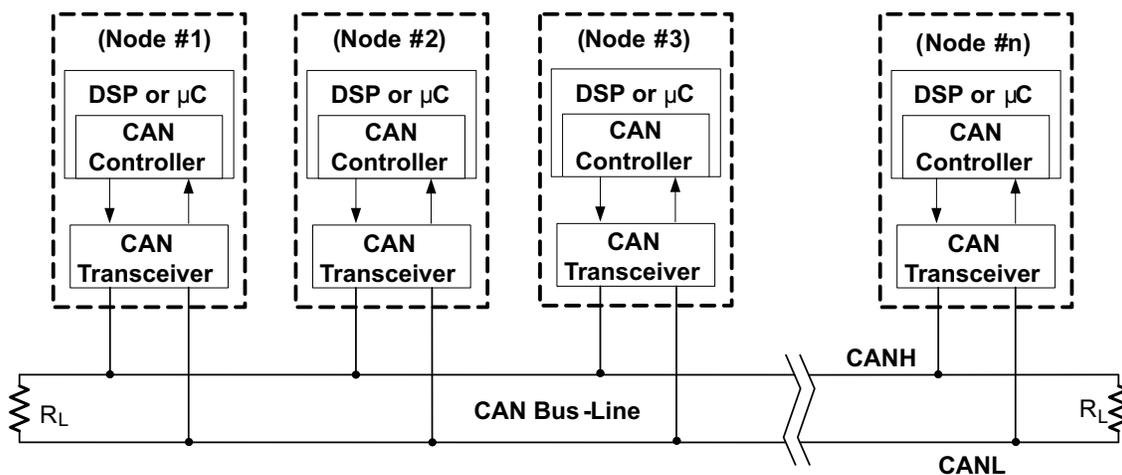
Figure 6. Details of a CAN Bus

Signaling is differential which is where CAN derives its robust noise immunity and fault tolerance. Balanced differential signaling reduces noise coupling and allows for high signaling rates over twisted-pair cable. Balanced means that the current flowing in each signal line is equal but opposite in direction, resulting in a field-canceling effect that is a key to low noise emissions. The use of balanced differential receivers and twisted-pair cabling enhance the common-mode rejection and high noise immunity of a CAN bus.

The High-Speed ISO 11898 Standard specifications are given for a maximum signaling rate of 1 Mbps with a bus length of 40 m with a maximum of 30 nodes. It also recommends a maximum unterminated stub length of 0.3 m. The cable is specified to be a shielded or unshielded twisted-pair with a 120-Ω characteristic impedance (Zo). The ISO 11898 Standard defines a single line of twisted-pair cable as the network topology as shown in Figure 6, terminated at both ends with 120-Ω resistors, which match the characteristic impedance of the line to prevent signal reflections. According to ISO 11898, placing R_L on a node must be avoided because the bus lines lose termination if the node is disconnected from the bus.

The two signal lines of the bus, CANH and CANL, in the quiescent recessive state, are passively biased to ≠ 2.5 V. The dominant state on the bus takes CANH ≠ 1 V higher to ≠ 3.5 V, and takes CANL ≠ 1 V lower to ≠ 1.5 V, creating a typical 2-V differential signal as displayed in Figure 7.
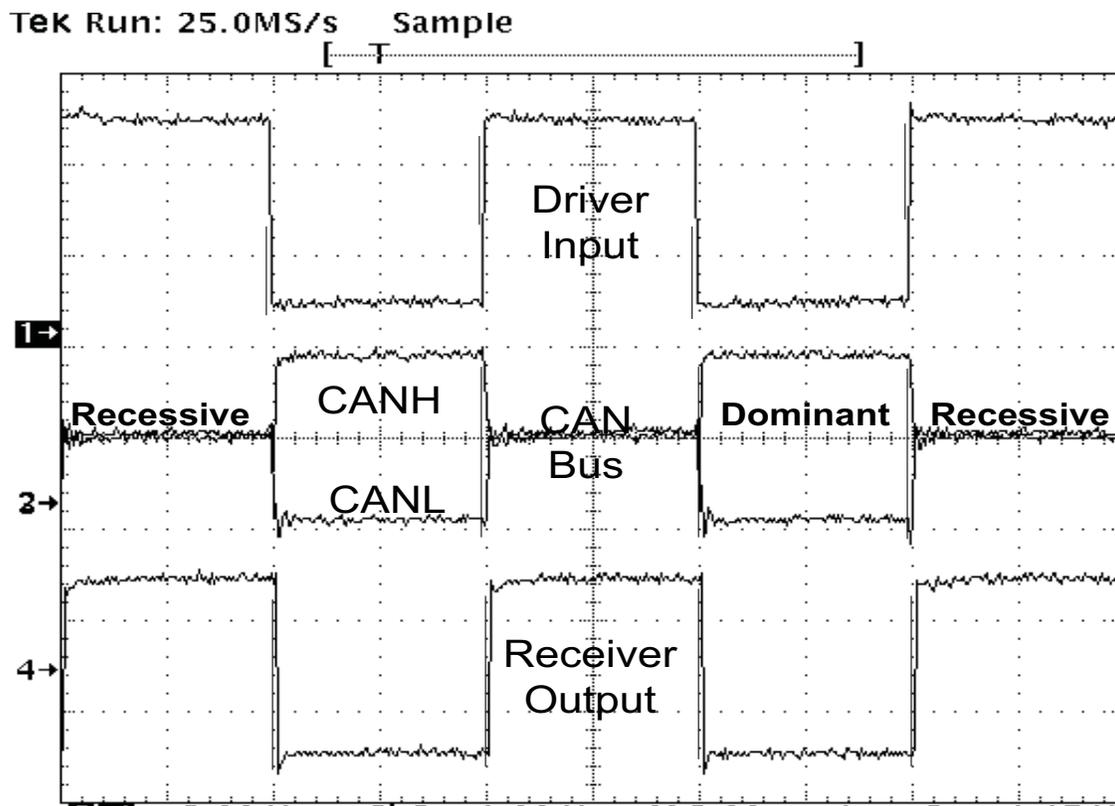
Figure 7. CAN Dominant and Recessive Bus States

The CAN standard defines a communication network that links all the nodes connected to a bus and enables them to talk with one another. There may or may not be a central control node, and nodes may be added at any time, even while the network is operating (hot-plugging).

The nodes in Figure 8 and Figure 9 could theoretically be sending messages from smart sensing technology and a motor controller. An actual application may include a temperature sensor sending out a temperature update that is used to adjust the motor speed of a fan. If a pressure sensor node wants to send a message at the same time, arbitration ensures that the message is sent.

For example, Node A in Figure 8 and Figure 9 finishes sending its message (on the left side of Figure 8) as nodes B and C acknowledge a correct message being received. Nodes B and C then begin arbitration—node C wins the arbitration and sends its message. Nodes A and B acknowledge C's message, and node B then continues on with its message. Again note the opposite polarity of the driver input and output on the bus.
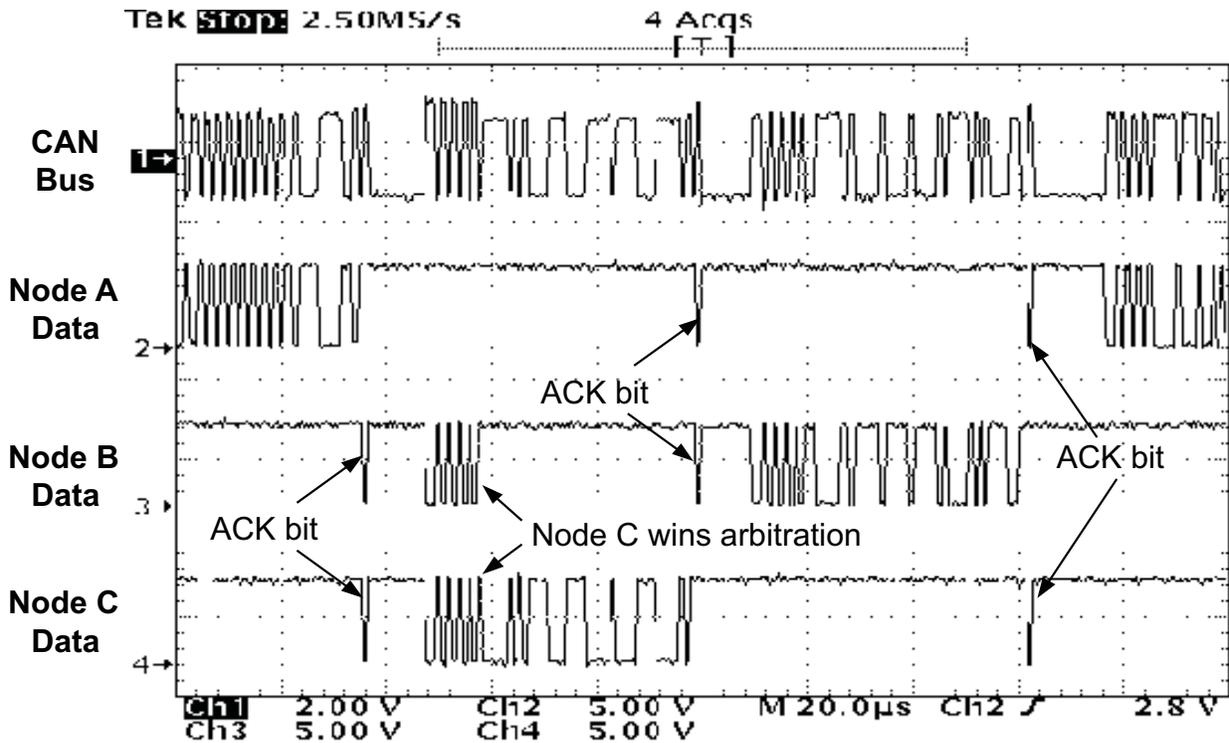
Figure 8. CAN Bus Traffic
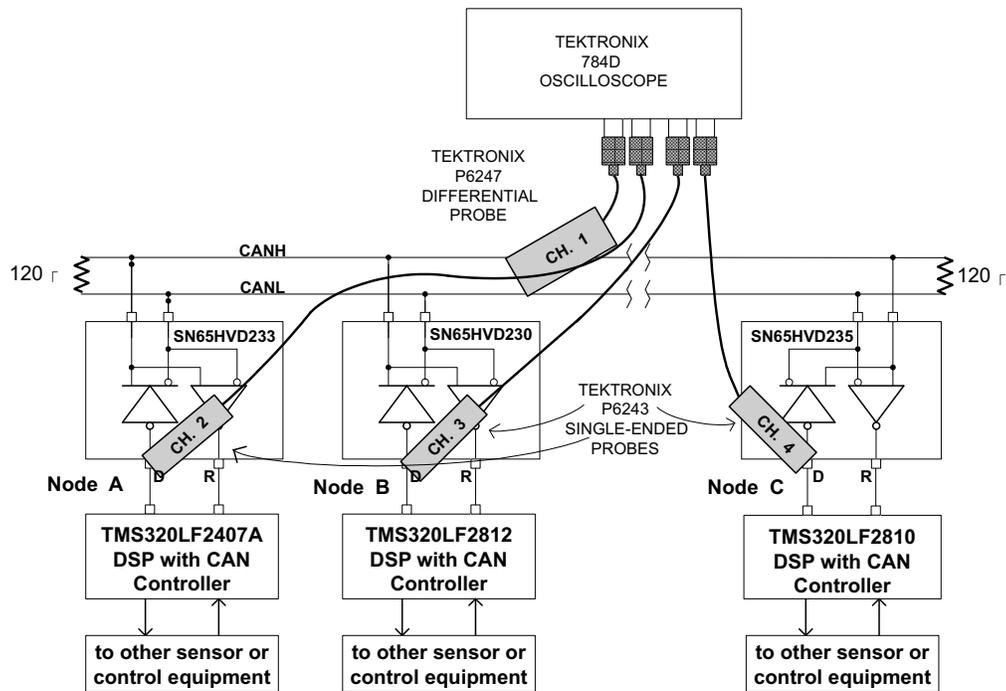
Figure 9. CAN Test Bus

5.1 CAN Transceiver Features

5.1.1 3.3-V Supply Voltage

Most CAN transceivers require a 5-V power supply to reach the signal levels required by the ISO 11898 standard. However, by superior attention to high-efficiency circuit design, the TI 3.3-V CAN transceiver family operates with a 3.3-V power supply and is fully interoperable with 5-V CAN transceivers on the same bus. This allows designers to reduce total node power by 50% or more (Figure 10).
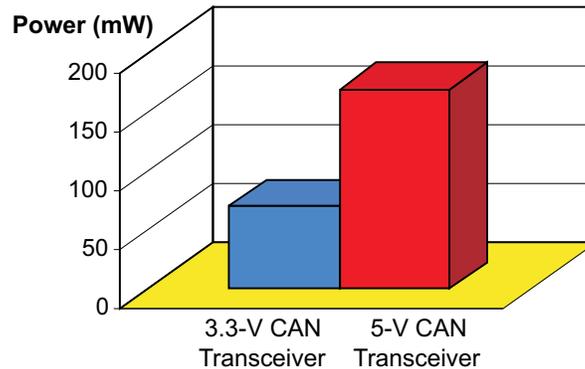
Figure 10. 3.3-V CAN Transceiver Power Savings

In addition to the inherent power savings of using a 3.3-V transceiver, for applications using 3.3-V technology, such as the TI TMS320C28xx family of DSPs with integrated CAN controllers, the need for a 5-V power supply can be eliminated. This lowers the overall part count for the node, reducing system cost and increasing system reliability.

For designers with an existing system design which requires a 5-V-powered transceiver, the TI 5-V transceivers are available with a wide variety of features such as high ESD protection and wide common-mode range.

5.1.2 ESD Protection

Static charge is an unbalanced electrical charge at rest, typically created by the physical contact of different materials. One surface gains electrons, whereas the other surface loses electrons. This results in an unbalanced electrical condition known as a static charge. When a static charge moves from one surface to another, it is referred to as an electrostatic discharge (ESD). It can occur only when the voltage differential between the two surfaces is sufficiently high to break down the dielectric strength of the medium separating the two surfaces.

ESD can occur in any one of four ways: a charged body can touch an integrated circuit (IC), a charged IC can touch a grounded surface, a charged machine can touch an IC, or an electrostatic field can induce a voltage across a dielectric sufficient to break it down.

The main threat of ESD damage occurs during the assembly and manufacturing of circuits. After assembly and installation, the main protection required for the bus pins is surge protection.

5.1.3 Common-Mode Voltage Operating Range

Common-mode voltage is the difference in potential between grounds of sending and receiving nodes on a bus. This is often the case in the networked equipment typically found in a CAN application. Possible effects of this problem are intermittent reboots, lock-ups, bad data transfer, or physical damage to a transceiver.

Network interface cards, parallel ports, serial ports, and especially transceivers are prime targets for some form of failure if not designed to accommodate high levels of ground shift and power supply imbalance between typical CAN nodes.

With this in mind, most TI CAN transceivers are designed to operate with complete safety well beyond the bus voltage range of –2 V to 7 V required by the ISO 11898 Standard

5.1.4 Common-Mode Noise Rejection

Common-mode noise of varied magnitudes exist within the networks associated with CAN applications. Noise from pulsing motor controllers, switch-mode power supplies, or from fluorescent lighting load are the typical sources of noises that couple onto bus lines as displayed in Figure 11. These would otherwise be straight lines.
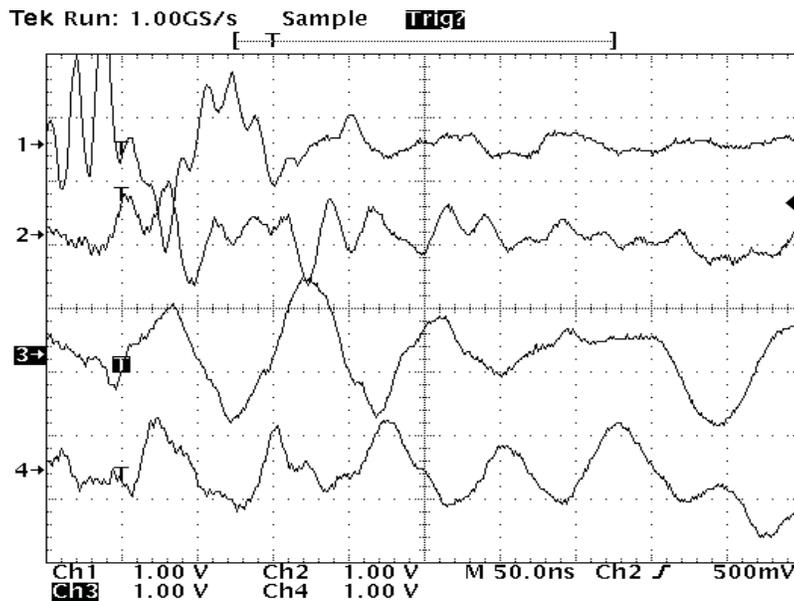
Figure 11. Common-Mode Noise Coupled Onto Four Twisted-Pair Bus Lines

A CAN transceiver's receiver not specifically designed to reject this coupled noise can respond to common-mode noise as if it were data on a bus and send meaningless data to a controller. TI CAN transceivers are specifically designed and tested for their ability to reject this common-mode noise.

5.1.5 Controlled Driver Output Transition Times

Controlling the driver output slew rate dampens the rise time of a dominant bit to improve signal quality and provides for longer stub lengths and a better bit-error rate. For a discussion on how slew-rate control provides for longer stub lengths, see application report SLLA270.

5.1.6 Low-Current Bus Monitor, Standby and Sleep Modes

Many applications are looking to lower-power opportunities as more electronics are added to designs. The standby mode in many TI transceivers is generally referred to as the "listen only" mode, because in standby, the driver circuitry is switched off while the receiver continues to monitor bus activity. In the occurrence of a dominant bit on the bus, the receiver passes this information along to its DSP/CAN controller which in turn activates the circuits that are in standby. This is achieved by placing a logic-low level on the Rs pin (pin 8) of the device.

The difference between the standby mode and the sleep mode is that both driver and receiver circuits can be switched off to create an extremely low-power sleep mode with no bus monitor. The local controller actively places the device into and out of sleep mode.

The HVD1040 contains the best of both standby and sleep features with a low-power (5 μA typical) bus monitor. The device driver and receiver circuitry is switched off while a small comparator monitors the bus and toggles the receiver output on bus activity.

5.1.7   Bus Pin Short-Circuit Protection

The ISO 11898 Standard recommends that a transceiver survive bus wire short-circuits to each other, to the power supply, and to ground. This ensures that transceivers are not damaged by bus cable polarity reversals, cable crush, and accidental shorts to high power supplies. The short-circuit protection in TI devices protects for an unlimited time. Once a problem is removed, the devices perform as designed whereas the CAN transceivers offered from competing vendors are permanently damaged and require replacement.

5.1.8   Thermal Shutdown Protection

Another desirable safety feature for a CAN transceiver is the thermal shutdown circuitry of TI CAN transceivers. This feature protects a device against the destructive currents and resulting heat that can occur in a short-circuit condition. Once thermal shutdown is activated, the device remains shut down until the circuitry is allowed to cool. Once cooled down to normal operating temperature, the device automatically returns to active operation without damage.

5.1.9   Bus Input Impedance

A high bus input impedance increases the number of nodes that can be added to a bus above the ISO 11898 Standard's 30 nodes. The high impedance restricts the amount of current that a receiver sinks or sources onto a bus over common-mode voltage conditions. This ensures that a driver transmitting a message into such a condition is not required to sink or source a damaging amount of current from the sum of the receiver leakage currents on a bus.

5.1.10   Glitch-Free Power Up and Power Down

This feature provides for hot-plugging onto a powered bus without disturbing the network. The TI driver and receiver pins are passively pulled high internally while the bus pins are biased internally to a high-impedance recessive state. This provides for a power up into a known recessive condition without disturbing ongoing bus communication.

5.1.11   Unpowered Node Protection

Many CAN transceivers on the market today have a low output impedance when unpowered. This low impedance causes the device to sink any signal present on the bus and shuts down all data transmission. TI CAN transceivers have a high output impedance in powered and unpowered conditions and maintain the integrity of the bus any time power or ground is removed from the circuit.

5.1.12   Reference Voltage

Reference voltage on a CAN transceiver is the Vref pin (pin 5) of what is considered to be the standard CAN transceiver footprint. This is the footprint of the first CAN transceiver to market, the NXP PCA82C250.

When first introduced, the Vcc/2 Vref pin served a particular NXP CAN controller as a voltage reference used to compare the bus voltage of a remaining single bus line in the event of an accident. If the voltages were the same, it was a recessive bit; if different, it was a dominant bit.

Although some users consider it handy for use as an actual voltage reference at the node, it is typically unused.

5.1.13   V-Split

V-split is a fortified Vcc/2 Vref pin with the same ESD protection rating, short-circuit protection, and common-mode operating range as the bus pins. It is used to stabilize bus voltage at Vcc/2 and prevent it from drifting to a high common-mode voltage during periods of inactivity.

It also filters unwanted high-frequency noise from bus lines with the termination technique of Figure 12. This is accomplished with a coupling capacitor between two ~60 Ω ±1% termination resistors to couple high-frequency noise to a solid ground potential. Care must be taken to match the two resistors carefully so as not to reduce the effective immunity. This technique improves the electromagnetic compatibility of a network. A typical value of CL for a high-speed CAN is 4.7 nF, which generates a 3-dB point at 1.1 Mbps. This, of course, is a signaling-rate-dependant value.

$$\text{Low-pass filter with } f_C = \frac{1}{2\pi R C_L}$$
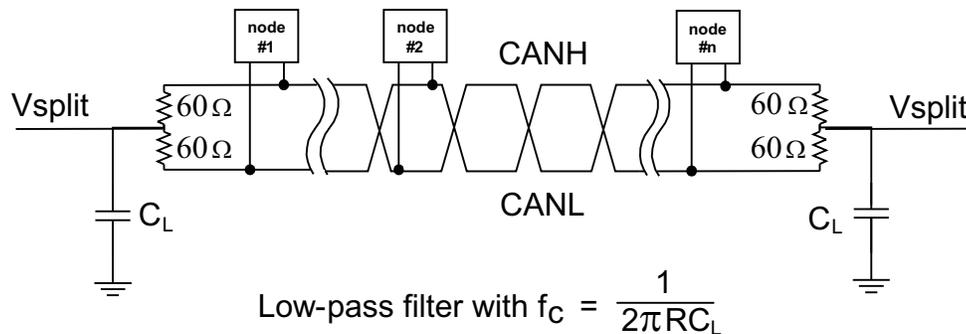
Figure 12. Split Termination

5.1.14 Loopback

This function places the bus input and output in a high-impedance state. The remaining transceiver circuitry remains active and available for driver-to-receiver loopback and self-diagnostic node functions without disturbing the bus.

5.1.15 Autobaud Loopback

In autobaud loopback, the "bus-transmit" function of the transceiver is disabled, while the "bus-receive" function and all of the normal operating functions of the device remain intact. With the autobaud function engaged, normal bus activity can be monitored by the device.

Autobaud detection is best suited to applications that have a known selection of baud rates. For example, a popular industrial application has optional settings of 125 kbps, 250 kbps, or 500 kbps. Once a logic-high has been applied to pin 5 (AB) of the HVD235, assume a baud rate such as 125 kbps; then wait for a message to be transmitted by another node on the bus. If the wrong baud rate has been selected, an error message is generated by the host CAN controller. However, because the "bus-transmit" function of the device has been disabled, no other nodes receive the error message of the controller.

This procedure makes use of the CAN controller's status register indications of message received and error warning status to signal if the current baud rate is correct or not. The warning status indicates that the CAN chip error counters have been incremented. A message-received status indicates that a good message has been received.

If an error is generated, reset the CAN controller with another baud rate and wait to receive another message. When an error-free message has been received, the correct baud rate has been detected.

5.2 CAN Transceiver Selection Guide

Transceiver	Supply Voltage (V)	Short Circuit Voltage (V)	Common-Mode Voltage Range (V)	ESD (kV)	Standby	Sleep	Silent	Shutdown (SHDN)	Low-Power Bus Monitor	Vref	Vsplit	Loopback	Autobaud	Slew Rate Control	Fault	VIO	5-Mbps FD
SN65HVD230	3.3	−4 to 16	−2 to 7	HBM: ±16	X					X				X			
SN65HVD231	3.3	−4 to 16	−2 to 7	HBM: ±16		X				X				X			
SN65HVD232	3.3	−4 to 16	−2 to 7	HBM: ±16													
SN65HVD233	3.3	±36	−7 to 12	HBM: ±16	X							X		X			
SN65HVD234	3.3	±36	−7 to 12	HBM:±16	X	X								X			
SN65HVD235	3.3	±36	−7 to 12	HBM: ±16	X								X	X			
SN65HVD251	5	±36	−7 to 12	HBM: ±12	X					X				X			
SN65HVD1040	5	−27 to 40	±12	IEC ±6 HBM: ±12					X		X						
SN65HVD1050	5	−27 to 40	±12	IEC: ±6 HBM: ±8	X						X						
SN65HVD255	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X								
SN65HVD256	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X							X	
SN65HVD257	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X						X		
SN65HVD265	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X								
SN65HVD266	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X							X	
SN65HVD267	5	−27 to 40	−2 to 7	IEC: ±8 HBM: ±12			X		X						X		
SN65HVDA540-5-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X												
SN65HVDA541-5-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X												
SN65HVDA542-5-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12			X										
SN65HVDA540-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X											X	
SN65HVDA541-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X											X	
SN65HVDA542-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12			X									X	
HVDA551-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X											X	
HVDA553-Q1	5	−27 to 40	±12	IEC: ±7 HBM: ±12	X						X						
TCAN330(G)	3.3	±14	±12	IEC ±12 HBM: ±4			X	X	X								X(G)
TCAN332(G)	3.3	±14	±12	IEC ±12 HBM: ±4													X(G)
TCAN334(G)	3.3	±14	±12	IEC ±12 HBM: ±4	X			X									X(G)

Transceiver	Supply Voltage (V)	Short Circuit Voltage (V)	Common-Mode Voltage Range (V)	ESD (kV)	Standby	Sleep	Silent	Shutdown (SHDN)	Low-Power Bus Monitor	Vref	Vsplit	Loopback	Autobaud	Slew Rate Control	Fault	VIO	5-Mbps FD
TCAN337(G)	3.3	±14	±12	IEC ±12 HBM: ±4			X		X						X		X(G)
TCAN1042(H)(G)(V)-Q1	5	±58 ±70 (H)	±30	IEC: ±8 HBM: ±10	X				X							X(V)	X(G)
TCAN1051(H)(G)(V)-Q1	5	±58 ±70 (H)	±30	IEC: ±8 HBM: ±10	X				X							X(V)	X(G)

6   Conclusion

CAN is ideally suited in applications requiring a large number of short messages with high reliability in rugged operating environments. Because CAN is message based and not address based, it is especially well suited when data is needed by more than one location and system-wide data consistency is mandatory.

Fault confinement is also a major benefit of CAN. Faulty nodes are automatically dropped from the bus, which prevents any single node from bringing a network down, and ensures that bandwidth is always available for critical message transmission. This error containment also allows nodes to be added to a bus while the system is in operation, otherwise known as hot-plugging.

The many features of the TI CAN transceivers make them ideally suited for the many rugged applications to which the CAN protocol is being adapted. Among the applications finding solutions with CAN are automobiles, trucks, motorcycles, snowmobiles trains, buses, airplanes, agriculture, construction, mining, and marine vehicles.

7   Additional Reading

1. Controller Area Network, Basics Protocols, Chips and Applications; Dr. Konrad Etschberger; ISBN 3-00-007376-0 (www.ixxat.com)
2. CAN Systems Engineering, From Theory to Practical Applications; Wolfhard Lawrenz, ISBN 0-387-94939-9

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.