Overview of Current Automotive Protocols

Version 1.1
2008-08-15
Application Note AN-AND-1-159

Restrictions	Public Document
Abstract	This application notes gives an overview of the major features of the most widely used low-level protocols in the automotive industry.

Table of Contents

1.0 Overview

Modern automotive electronic applications are implemented over distributed architectures, which may include several electronic control units (ECUs) communicating via one or more communication protocols over network buses. Serial multiplex communication (often written as "MUX" by automotive engineers) is a method of reducing wiring requirements while increasing the amount and type of data to be shared between various components in the automobile. This is done by connecting each component (or *node*), to a serial data bus. Each node collects whatever data is useful to both it and other nodes (wheel speed, engine RPM, oil pressure, etc.), and then transmits this data onto the bus, where any other node needing this data can receive it.

There are several major protocols in use today that are competing for the automotive market. While some protocols are essentially "internal corporate" solutions, several of these protocols have emerged to become either ISO (International Standards Organization) or SAE (Society of Automotive Engineers) industry-level recommended practices or standards. This application note introduces these current protocols.

Contact Information: www.vector-cantech.com or 1-248-449-9290

2.0 Current Automotive Protocols

The following table gives an overview of the major small area networking protocols in use today in the automotive industry. This list does not claim to be complete. Others may be added to this document as needed.

Protocol	Description	Used By
CAN	Controller Area Network	Almost all car companies
FlexRay	High speed time triggered serial communication bus	BMW AG, Daimler, Chrysler, Fiat, Ford, GM
J1850	Being phased out	DC, Ford, GM
LIN	Local Interconnect Network	BMW, DC, Fiat, Ford, GM, Honda, Nissan, Peugeot, Renault, Toyota, VW
MOST	Media Oriented Systems Transport	BMW, DC, Fiat, Ford, GM, Honda, Nissan, Peugeot, Toyota, VW
TTCAN	Time-Triggered CAN	Adoption level for automotive is near zero
TTP	Time-Triggered Protocol	Peugeot, Renault, VW

Table 1 – Data Bus Implementations by Major OEMs (from Automotive Engineering)

The CAN protocol has three major bus implementations:

Protocol	Description	Used By
HS-CAN	High-Speed CAN	BMW, DC, Fiat, Ford, GM, Hyundai, Nissan, Peugeot, Renault, Toyota, Volkswagen
FT-CAN	Fault-Tolerant CAN	BMW, DC, Fiat, Hyundai, Nissan, Peugeot, Renault, Volkswagen
SW-CAN	Single-Wire CAN	GM, Honda

Table 2 – CAN Protocol Implementations

2.1 CAN

Controller Area Network (CAN) is a serial communications protocol that supports real-time control while maintaining a high level of data integrity. Originally developed for automotive subsystem applications, CAN has gained wide acceptance in a number of application areas. CAN is an open international standard which is defined in the CAN 2.0B specification maintain by the industry organization known as “CAN In Automation”, or CIA.

CAN is a message oriented transmission protocol. Messages are identified by a message identifier. Each identifier must be unique within the network, as it defines the identity and the priority of the message. This is important when several nodes are competing for access to the bus.

The priority of each message is specified by the message identifier. The identifier with the lowest binary number has the highest priority. Bus access conflicts are resolved by bit-wise arbitration with the identifiers involved by each node observing the bus level bit for bit with the "wired and" mechanism, by which the dominant state overwrites the recessive state. All nodes with recessive transmission and dominant observation lose the competition for bus access. They automatically become receivers of the message and do not re-attempt transmission until the bus is available again.

Transmission requests are handled in the order of importance of the messages for the system as a whole, which is helpful in overload situations. Since bus access is prioritized on the basis of the messages, low individual latency times in real-time systems can be guaranteed.

More information about the CAN specification can be found at www.can-cia.de .

There are three major variants of CAN being implemented today in the Physical Layer: High-Speed CAN, Fault-Tolerant CAN and Single-Wire CAN.

2.1.1 High-Speed CAN

CAN high-speed physical layers are most often defined by the ISO 11898-2 and SAE J2284 standards, both of which define requirements for a high-speed, dual-wire bus line with common return terminated at both ends by resistors to suppress reflections representing the characteristic impedance of the line. This physical layer is also compatible with other protocols that are based on dominant/recessive logic. It is intended for applications where a high data communication rate is the main requirement.

2.1.2 Fault-Tolerant CAN

Fault-tolerant CAN is a physical layer that transmits at medium speed (single transfer rate limited to a maximum of 125K BPS) for body bus applications.

Fault tolerance is the ability of a system or component to continue normal operation despite the presence of hardware or software faults. A fault tolerant CAN transceiver can continue communication after one single network wiring problem. Communication will be possible for any of the following single wiring faults on the CAN bus: the CAN_H wire is open, the CAN_L wire is open, a short circuit between CAN_H and ground, a short circuit between CAN_H and battery (+12V), a short circuit between CAN_L and ground, a short circuit between CAN_L and battery (+12V), or a short circuit between CAN_H and CAN_L.

There are different fault-tolerant CAN physical layer standards. The ISO 11519-2 standard describes transceiver circuitry for older heavy duty truck & bus applications. This standard will be withdrawn very soon. The physical layer specified in ISO-11992 is for truck/trailer applications providing high differential voltage. The ISO TC22 SC3 WG1 standardization group has published an International Standard 11898-3 for fault-tolerant transceivers, which is to replace ISO 11519-2. This specifies low-power consumption and switch-off modes as well as the wake-up procedure.

Nearly all carmakers in Europe, America, and the Far East use CAN high-speed networks (e.g. 500 kbit/s) in their power engine systems, which are compliant with the ISO 11898-2 physical layer standard. In addition, most European passenger cars have CAN-based multiplex systems to link door and roof control units as well as lighting and seat control units. Some of the multiplex networks are compliant to the fault-tolerant physical layer standard (ISO 11898-3).

2.1.3 Single-Wire CAN

Known industry-wide as SAE J2411, Single-Wire CAN (also called SWCAN) establishes requirements for a low speed, single wire, physical layer with sleep/wakeup capability. J2411 is compatible with CAN and all other protocols that are based on dominant/recessive logic.

Single-wire CAN is used in body electronics applications such as climate control, door locks, instruments clusters, seat positioning and other body and convenience systems. It can also be used for flash programming.

The single wire CAN Physical Layer contains three operational modes: 1) normal communication mode, 2) high-voltage wake up mode, and 3) high-speed mode.

J2411 establishes two data rates: a normal rate of 33.3 kbit/s and a maximum of 32 nodes and a high-speed rate of 83.3 kbit/s.

2.2 FlexRay

FlexRay is a bus system intended for high-speed applications in automotive engineering. The protocol targets x-by-wire systems applications that demand high-speed bus systems that are deterministic, fault-tolerant, and capable of supporting distributed control systems.

FlexRay was developed by several automotive OEMs and suppliers in cooperation with semiconductor manufacturers. Its features include:

- Static and dynamic data transmission (scalable)
- Gross data rate of up to 20 Mbit/sec (2 channels with 10 Mbit/sec each)
- Time-triggered services implemented in hardware

FlexRay uses a specifically designed high-speed transceiver, and it embraces the definition of hardware and software interfaces between various components of a FlexRay node. The FlexRay protocol defines the format and function of the communication process within a networked automotive system.

FlexRay is initially targeted for a data rate of approximately 2.5, 5 and 10Mbit/sec per channel, but the design of the protocol allows much higher data rates.

Depending on the application needs, the communication cycle can be purely static, purely dynamic, or a mixture of both. Static data transmission enables time-triggered communication. Dynamic transmission allows each node to use the full bandwidth for event-driven communication.

More information about FlexRay can be found at www.flexray.com.

2.3 J1850

SAE Recommended Practice J1850 was developed independently by North American automotive OEMs as a medium speed (Class B) serial multiplex communication bus for use in the automotive environment. Used primarily during the 1990's, J1850 has almost completely been phased out and has been primarily replaced by CAN.

J1850 attributes include an open architecture, asynchronous messages, single-level bus topology and serial multiplex communication.

The J1850 protocol encompasses the lowest two layers of the ISO open system interconnect (OSI) model, the Data Link Layer and the Physical Layer. It is a multi-master system, utilizing the concept of carrier sense multiple access with collision resolution (CSMA/CR), where any node can transmit if it has determined that the bus is free. Non-destructive arbitration is performed on a bit-by-bit basis whenever multiple nodes begin to transmit simultaneously.

A J1850 message, or frame, consists of a start of frame (SOF) delimiter, a one- or three-byte header, zero to eight data bytes, a CRC or Checksum byte, an end of data (EOD) delimiter, and an optional in-frame response byte, followed by an end of frame (EOF) delimiter.

J1850 allows for the use of a single or dual wire bus, two bit encoding techniques (pulse-width modulation or variable pulse-width modulation), and the use of CRC or Checksum for error detection, depending upon the message format and modulation technique selected. J1850 also allows for two data rates. Ford's version of J1850 runs at 41.7 kbps over a two-wire differential pair using conventional PWM modulation. By contrast, GM's version runs at 10.4 kbps on a single wire using a variable pulse width scheme.

J1850 applications include body control subsystems, engine management, transmission, ABS, and instrumentation.

2.4 LIN

LIN, or Local Interconnect Network, is a low-speed (1 to 20 kilobits per second) serial multiplexing protocol primarily intended for body electronics systems in vehicles (e.g., seat controls). Applications target the interconnection of switches, actuators and sensors into a localized sub-bus which connects to the main bus, which is usually a CAN bus. Vehicle subsystems that could use a LIN sub-bus are the door, roof, steering column, climate control, switch panel and intelligent wipers. LIN can also be used for engine diagnostics. LIN, however, is not designed exclusively for auto applications, and so can be applied to industrial electronics as well.

The LIN Consortium (the LIN standard organization) started as a work group in late 1998. The objective of this work group is to specify an open standard for low-cost Local Interconnect Networks (LINs) in vehicles where the bandwidth and versatility of CAN are not required.

The governing LIN Protocol Specification establishes the key LIN Data Link Layer requirements. Basic LIN data link transfers are based on the UART (Universal Asynchronous Receiver Transmitter), a common serial communication method available on many microcontrollers. The LIN standard includes the specification of the transmission protocol, the transmission medium, the interface between development tools, and the interfaces for software programming.

More information about LIN can be found at www.lin-subbus.org.

2.5 MOST

The MOST (Media Oriented Systems Transport) fiber optic network was originally designed by a semiconductor manufacturer in cooperation with automotive OEMs and mobile entertainment suppliers. It provides mechanisms for transporting high volumes of streaming multimedia data. Typical applications are high-end audio transmission, the navigational system and other infotainment devices in the car.

A MOST network, usually laid out as a ring, may include up to 64 MOST devices. One MOST device handles the role of the Timing Master, which continuously feeds MOST frames into the ring. The preamble, sent out at the beginning of the transmission of a MOST frame, allows the Timing Slaves to synchronize. The Timing Slaves can continuously resynchronize themselves to the bi-phase coding that underlies the synchronous transmission.

In total, the bandwidth available for transmitting streaming data (synchronous data transmission) and packet data (asynchronous data transmission) is approx. 23 MBaud. This is subdivided into 60 physical channels which the user can select and configure. MOST provides numerous services and mechanisms for the allocation (and deallocation) of physical channels.

MOST supports up to 15 uncompressed stereo audio channels of CD quality or up to 15 MPEG1 channels for audio-video transmission. However it is still not possible to transmit high-resolution uncompressed video data streams with MOST.

Simultaneously MOST provides a channel for transmitting control messages. A bandwidth of 768 kBaud is available for this purpose, or in other words almost 3000 control messages can be transmitted per second. These control messages can be used to configure MOST devices as well as to synchronously and asynchronously transmit data. Various function blocks are specified by the MOST cooperation for this purpose, which describe the static interface of applications in order to enable the interoperability of devices.

Recent developments include MOST50 over twisted pair with a doubled bandwidth compared to the original MOST and even MOST150 with a tripled bandwidth.

MOST is used by several auto companies. Ford uses it only in their Premier Automotive Group (PAG), i.e. in the Land Rover, Jaguar and Volvo. As for GM, only SAAB has used MOST, but have now discontinued using it. Audi and Porsche use MOST; Nissan and Honda do not.

More information about MOST can be found at http://www.mostnet.org.

2.6 TTCAN

ISO (International Standardization Organization) has specified in ISO 11898-4 the TTCAN (time-triggered communication on CAN) protocol. It is a communication option for CAN-based networks that describes the prerequisites needed to synchronize all nodes in a CAN network. When the nodes are synchronized, any message can be transmitted in a specific time slot without having to compete with other messages for the bus. This avoids loss of arbitration, and the latency time becomes predictable. Aside from the synchronized communication schedule, TTCAN nodes operate according to the standard CAN protocol as defined by ISO 11898-1.

A common time reference or global system time is provided to synchronize the activities of all CAN nodes in a network. Each node has its own local time, which is a counter that is incremented each network time unit (NTU). The system wide NTU is derived from the node's local clock and local time unit ratio (TUR). In TTCAN, the cyclic transmission schedule is synchronized by the repeated transmission of a particular CAN message called the reference message. This reference message (transmitted by a time master) restarts the cycle time in each node. The cycle time is derived from the node's local time.

Bosch has implemented the TTCAN module in a chip. The TTCAN stand-alone chip with integrated Level 1 and Level 2 (with time correction) is pin-compatible to the company's CAN controller CC170 and Intel's 82527. It is available for evaluating TTCAN interfaces and developing TTCAN-compatible ECUs and software tools (e.g. bus analyzers).

Atmel offers CAN controllers with partial TTCAN support. The 8-bit T89C51CC01/2 provides 15 message objects that are adjustable via filter masks. The message objects may optionally be transformed to a FIFO, which prevents the erasure of received but not yet processed data.

The NEC CAN microcontroller family provides the hardware requirements for TTCAN. NEC is working on a hardware version of a standalone TTCAN module. Hitachi has implemented the protocol in a chip, which is being tested by an independent institute for conformity to the standard.

CiA members are currently working on extending the CANopen standard in order to enable the usage of TTCAN hardware within CANopen networks. The CANopen extensions are due for publication with the official release of the TTCAN standard.

More information on TTCAN can be obtained from the Robert Bosch Corporation at www.can.bosch.com, from the CiA at www.can-cia.org, and from TTCAN's own website (www.ttcan.com).

2.7 TTP

TTP, or Time-Triggered Protocol, supports soft real-time or event-triggered data traffic, such as remote access and advanced diagnosis strategies. A time-division multiple access bus acts as the basic functional principle of TTP. A distributed algorithm establishes the global time base with a steady clock synchronization. It integrates such services as predictable message transmission, clock synchronization, membership, mode change, and blackout handling. It also supports replicated nodes and replicated communication channels.

TTP has been developed with the objective of achieving an open standard for fault tolerant systems. Leading research institutes and corporations have been involved in the development of TTP over the last 20 years.

Unlike communication systems such as a CAN bus, which run asynchronously and are driven by events, in TTP-based systems all connected nodes, for example brakes and steering, communicate on the basis of specified time intervals, while avoiding any data collision. They also communicate over redundant data busses. The precision of the time intervals is in the microsecond range.

As a complement to TTP, the Time-Triggered Architecture (TTA) provides a computing infrastructure for the design and implementation of distributed embedded systems. A large real-time application is decomposed into nearly autonomous clusters and nodes. A fault-tolerant global time base of known precision is generated at every node to precisely specify the interfaces among the nodes, to simplify the communication and agreement protocols, and to perform prompt error detection.

The company TTTech provides a complete and harmonized tool chain including interfaces to third-party software and automation functionality. They may be contacted at www.tttech.com.

2.7.1 TTP/C

TTP™/C is an integrated communication protocol for hard real-time fault-tolerant distributed systems.

TTP/C is member of the TTP protocol family. The “C” indicates that it satisfies SAE (Society of Automotive Engineers) Class C requirements for communication in the automotive area. Class C protocols are suitable for high-speed, single-failure operational safety-critical applications. The protocols currently used by automotive engineers, such as the J1850 family and CAN, are suitable only for Class A and Class B systems, which are subject to less rigorous requirements. TTP/C is the first protocol designed to meet the additional safety-critical requirements of Class C. The first class of applications in development using TTP/C are the ‘X-by-wire’ applications, such as brake-by-wire, in which all-hydraulic and mechanical systems are replaced by electronics.

TTP/C provides hard real-time message delivery with little jitter and distributed fault-tolerant clock synchronization. Various fault-tolerance strategies are supported, so that no single failure of any part of the communication system should disrupt communication. Extensive mechanisms for error detection, recovery and re-integration of nodes are provided.

TTP/C can also be used for time- and event-triggered data transmission. To do so, dedicated bandwidth has to be allocated per-node for event-triggered activities. The purpose is to ensure that event or diagnostic data has no influence over safety-critical, real-time data. For event-triggered data, TTP/C uses the same identifiers as any other protocol, and the event-triggered data can only be sent inside time slots, which are non-conflicting. To prevent the “babbling idiot mode,” TTP/C provides an independent bus guardian, which ensures exclusive access to the bus.

The Communication Network Interface (CNI) acts as a temporal firewall to guarantee that none of the host computers can influence the temporal pattern of the communication system. With TTP/C the communication controller has its own memory and data structures—the MEDL (message descriptor list)—that defines all the protocol activities such as the points in time when to send and receive messages. The temporal coupling between the communication controller and the host is thus restricted to a minimum. The host CPU can read and write messages from the communication controller via a simple dual port RAM interface.

3.0 Contacts

Vector Informatik GmbH
Ingersheimer Straße 24
70499 Stuttgart
Germany
Tel.: +49 711-80670-0
Fax: +49 711-80670-111
Email: info@vector-informatik.de

Vector CANtech, Inc.
39500 Orchard Hill Pl., Ste 550
Novi, MI 48375
USA
Tel: +1-248-449-9290
Fax: +1-248-449-9704
Email: info@vector-cantech.com

VecScan AB
Theres Svenssons Gata 9
41755 Göteborg
Sweden
Tel: +46 (0)31 764 76 00
Fax: +46 (0)31 764 76 19
Email: info@vecscan.com

Vector France SAS
168 Boulevard Camélinat
92240 Malakoff
France
Tel: +33 (0)1 42 31 40 00
Fax: +33 (0)1 42 31 40 09
Email: information@vector-france.fr

Vector Japan Co. Ltd.
Seafort Square Center Bld. 18F
2-3-12, Higashi-shinagawa,
Shinagawa-ku
Tokyo 140-0002
Japan
Tel.: +81 3 5769 7800
Fax: +81 3 5769 6975
Email: info@vector-japan.co.jp

Vector Korea IT Inc.
Daerung Post Tower III, 508
182-4 Guro-dong, Guro-gu
Seoul, 152-790
Republic of Korea
Tel.: +82-2-2028-0600
Fax: +82-2-2028-0604
Email: info@vector-korea.com