In-Vehicle Networking

Automotive Controller Area Network (CAN) Applications

Overview

The Controller Area Network (CAN) is a serial, asynchronous, multi-master communication protocol for connecting electronic control modules in automotive and industrial applications. CAN was designed for automotive applications needing high levels of data integrity and data rates of up to 1 Mbit/s. Freescale Semiconductor has a complete line of products to enable automotive electronics designers to incorporate CAN into their applications.

Key Benefits

> The automotive networking standard protocol for Europe

> Supports the US automakers migration to CAN for body electronics busses

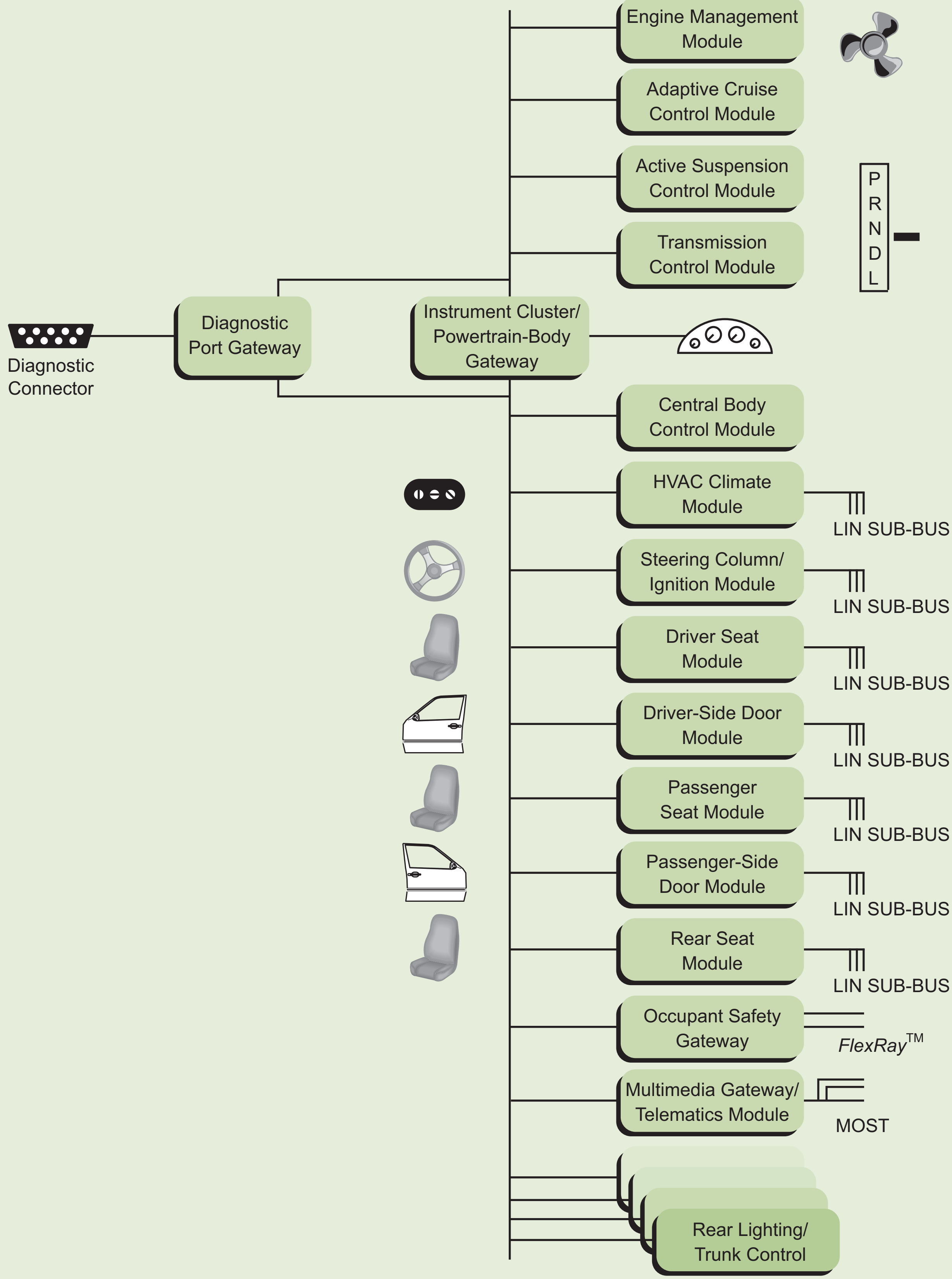
> Provides plentiful and proven Freescale Semiconductor CAN products and tools

> Exists in all levels of Freescale Semiconductor microcontrollers and DSPs

> Provides connectivity and increased integration using Freescale Semiconductor *SMARTMOS*™ CAN physical layers and System Basis Chips

freescale™ semiconductor

AUTOMOTIVE CAN NETWORKS



SG2032-2

Freescale Ordering Information[Note]

Part Number	Product Highlights	Additional Information
Analog Devices		
MC33388	Fault Tolerant CAN Interface	www.freescale.com/analog
MC33389	System Basis Chip with Low-Speed CAN	
MC33399	Local Interconnect Network (LIN) Physical Layer	
MC33661	eLIN – Enhanced LIN Physical Layer (Local Interconnect Network)	
MC33689	System Basis Chip with Enhanced LIN Physical Interface	
MC33742	System Basis Chip with Enhanced High-Speed CAN	
MC33889	System Basis Chip with Low-Speed Fault Tolerant CAN	
MC33897	Single-Wire CAN Transceiver	
MC33989	System Basis Chip with High-Speed CAN	
DSP Devices		
DSP56F803BU80	16-Bit (DSP/MCU) with 1 MSCAN12 Module	www.freescale.com
DSP56F805FV80	16-Bit (DSP/MCU) with 1 MSCAN12 Module	
DSP56F807PY80	16-Bit (DSP/MCU) with 1 MSCAN12 Module	
DSP56F807VF80	16-Bit (DSP/MCU) with 1 MSCAN12 Module	
HC08		
HC08 Family	Up to 60 K of Flash or ROM Memory; Enhanced SCI for LIN; SPI; Clock Generation Module; Freescale Semiconductor Scalable CAN	
MC68HC908AZxx Family	1 MSCAN08 Module	
HC12		
HC12 Family	Up to 128 K of Flash or ROM; SCI; SPI; Clock Generation Module; Up to Three CAN Modules	
XC68HC912BCxx Family	1 MSCAN12 Module	
MC68HC912Dxx(A) Family	2 MSCAN12 Modules	
MC68HC912DG128A	2 MSCAN12 Modules	
MC68HC912DT128A	3 MSCAN12 Modules	
HCS12		
HCS12 Family	Up to 512 K of Flash or ROM; Up to Two ESCI; Up to Three SPI; Up to 4 CAN Modules; Clock Generators; Excellent EMC and Stop Idd	
MC9S12DGxx Family	2 MSCAN12 (rev. 2.0) Modules	
MC9S12DJxx Family	2 MSCAN12 (rev. 2.0) Modules, 1 BDLC (J1850) Module	
MC9S12DPxx Family	5 MSCAN12 (rev. 2.0) Modules	
MC9S12DTxx Family	3 MSCAN12 (rev. 2.0) Modules	
MC9S12Dxx Family	1 MSCAN12 (rev. 2.0) Module	
MC9S12Hxx Family	2 MSCAN12 (rev. 2.0) Modules	
MC9S12Cxx Family	1 MSCAN12 (rev. 2.0) Module	
32-Bit Microcontrollers		
MPC555/6LFMZP40(R2)	2 TouCAN Modules	
MPC561/2LFMZP40(R2)	3 TouCAN Modules	
MPC563/4LFMZP40(R2)	3 TouCAN Modules	
MPC565/6LFMZP40(R2)	3 TouCAN Modules	
MPC5200 32-bit Processors		
MPC5200	2 MSCAN12 2.0a/2.0b	e-www.freescale.com/files/abstract/ overview/SPSMPC5200.htm

Note: Search on the listed part number.

Design Challenges

Different CAN Networks Have Different Performance Needs

Not all CAN networks are created equal. In the automotive environment, CAN networks can be split into two distinct categories based on the nature of the traffic on the network. Body control networks, dealing with passenger comfort and convenience systems for example, deal with a wide range of message identifiers that appear in no particular order or frequency. In contrast, powertrain networks that pass information relating to engine and transmission control have a much lower number of different messages to deal with, but the messages appear very rapidly and very regularly. These differences in messaging result in very different approaches to designing hardware and software systems to deal with the demands each type of network places on each node in that network.

Different CAN Networks Have Physical Layer Requirements

CAN, like all major networking protocols, requires a physical layer device to communicate. This physical layer comes from the ISO/OSI seven layer stack model and is responsible for current and voltage control for the bus. It deals with current and voltage transients and signalling bus (line) faults and works to possibly correct them.

The Bosch CAN specification does not dictate physical layer specifications for anyone implementing a CAN network. This is both a blessing and a curse to the designer. Over the course of the last decade, two major physical layer designs have come to the forefront and become the basic physical layer designs used in most CAN applications. They both communicate using a differential voltage on a pair of wires and are commonly referred to as a high-speed and a low-speed physical layer. The low-speed architecture has the ability to change to a single-wire operating architecture (referenced off ground) when one of the two wires is faulted through a short or open. Because of the nature of the circuitry required to perform this function, this architecture is very expensive to implement at bus speeds above 125 kbit/s. This is why 125 kbit/s is the dividing line between what is considered high-speed and low-speed CAN. Although both architectures use a voltage difference on a pair of wires, the termination methods for each are different and incompatible in production systems.

One additional CAN physical layer has recently been developed by General Motors. This physical layer uses only one wire at all times that limits its speed performance to 33.33 kbit/s. This single-wire CAN physical layer is very different from the other two types and is not yet widely accepted.

Because there are no requirements on the physical layer in the CAN specification, other standards organizations have developed standards to help designers create compatible CAN devices. The International Standards Organization (ISO) and Society of Automotive Engineers (SAE) create the standards for Europe and the United States respectively, to ensure interoperability of components at the physical layer and recommended design practices.

Automotive CAN Standards in Europe and the United States

CAN Physical Layer Type	ISO Standards (Europe) www.iso.org	SAE Standards (North America) www.sae.org
Single-Wire CAN	n/a	SAE J2411 Single Wire CAN Network for Vehicle Applications
Low-Speed Fault-Tolerant CAN	ISO 11519-2 Road Vehicles—Low-Speed Serial Data Communication— Part 2: Low-Speed Controller Area Network (CAN) ISO 11898-3 may be replaced with 11519-2	The ISO standard is generally used
High-Speed CAN	ISO 11898 Road Vehicles—Interchange of Digital Information—Controller Area Network (CAN) for High-Speed Communication	SAE J2284-125, SAE J2284-250, and SAE J2284-500 High Speed CAN (HSC) for Vehicle Application at 125 kbps, 250 kbps, and 500 kpbs, respectively
Local Area Network LIN	LIN Specification	LIN Specification

Freescale Semiconductor Solution

Different CAN Implementations to Meet Different CAN Performance Needs

Freescale Semiconductor recognizes the challenges that face designers of automotive CAN devices and systems. Because different CAN networks have different CAN messaging needs, Freescale Semiconductor provides different CAN hardware options to meet these challenges.

Freescale Semiconductor's 32-bit microcontrollers use either the *TouCAN™* or *FlexCAN* ™ hardware modules to communicate on the CAN bus. These modules are based on the traditional mailbox or "full-CAN" hardware architecture that provides 16 message buffers. When messages are received, a hardware filter match will drop the message into one of the 'mailboxes' (receive buffers). This approach works very well with powertrain systems, where messages are very regular and predictable, as the application designer can ensure that the software empties the mailboxes fast enough to keep new messages from over-writing the old messages in the boxes. If the boxes are not emptied fast enough because multiple messages come in too quickly with the same identifier, data can be lost. This is why mailbox architectures are not always suited to networks with unpredictable, event-driven data.

As stated before, body electronics networks have messaging that can be very sporadic and unpredictable in nature, which makes the Freescale Semiconductor Scalable CAN (msCAN) architecture so well suited to these applications. Since HC08, HC12, and HCS12 microcontroller families are 8-bit and 16-bit microcontrollers which are the backbone of body electronics systems and components, the msCAN module is a perfect fit for these families. CAN messages received by msCAN are placed into a single first-in, first-out (FIFO) storage structure. This structure maintains the order of received messages and allows many messages with identical identifiers to be received in rapid succession without concern of the overflow of a single receive buffer.

Freescale Semiconductor SMOS CAN Physical Layer Products to Meet Automotive Customer Needs

To address the need for multiple types of CAN physical layers, Freescale Semiconductor offers a range of CAN physical layer devices designed to meet or exceed the performance standards set out by ISO and SAE.

But a simple physical layer device is not always enough. For example, all automotive modules need to run from a regulated power supply. Sometimes a local switch or sensor might need to wake up the module from sleep state to active running state very quickly. That switch or sensor might be running at vehicle battery levels. This is where the Freescale Semiconductor System Basis Chip (SBC) brings power and value to the automotive design table. SBCs combine the CAN physical layers needed for automotive CAN connectivity with voltage regulation, independent watchdog timer, and local wake-up circuitry to allow greater flexibility with fewer components. Since these circuits can be made with the same semiconductor processes, it makes sense to combine these functions into one package and reduce the number of components needed in the final design. This reduces assembly costs, increases reliability, and increases design flexibility.

Development Tools [Note]

Tool Type	Product Name	Vendor	Description	Additional Information
Software drivers	MSCAN Low-Level Software Drivers	Metrowerks	Low-Level Driver Software for MSCAN08, MSCAN12, and MSCAN for HCS12	www.metrowerks.com
Configuration tool	MSCAN Filter Generation Tool	Metrowerks	Calculates Optimal Hardware Filter Settings for MSCAN Architecture for Customer Application	
Hardware development tools	EVBs and Other Development Tools for Respective MCUs and Analog Devices	Metrowerks	Helps Developers Simplify and Speed Development of High-Performance Microcontrollers	
Evaluation Kit	KIT33388DEVB	Metrowerks	Fault Tolerant CAN Interface	
Evaluation Kit	KIT33389DWEVB	Metrowerks	System Basis Chip	
Evaluation Kit	KIT33399DEVB	Metrowerks	Local Interconnect Network (LIN) Physical Layer	
Evaluation Kit	KIT33661DEVB	Metrowerks	LIN Enhanced Physical Interface	
Evaluation Kit	KIT33689DWBEVB	Metrowerks	System Basis Chip with LIN Transceiver	
Evaluation Kit	KIT33742DWEVB	Metrowerks	System Basis Chip with Enhanced High-Speed CAN	
Evaluation Kit	KIT33889DWEVB	Metrowerks	System Basis Chip with Low-Speed CAN	
Evaluation Kit	KIT33989DWEVB	Metrowerks	System Basis Chip with High-Speed CAN	

Note: Search on the listed product name.

Third Party Support		
Product Name	**Description**	**Contact Method**
Vector CANtech		
CANalyzer	CAN Network Analysis and Development Tool	www.vector-cantech.com
CANoe	CAN System Level Message Analysis and Modeling for Multiple Modules	
CANape Graph	ECU Monitor and Calibration Tool Using CAN and CCP	
CANscope	Digitized Oscilloscope of CAN Message Wave Forms	
SW drivers for Freescale Semiconductor MCUs for GM, Ford, and DaimlerChrysler	Software Drivers	
Volcano Automotive Group		
VNA - Volcano Network Architect	Stand-alone offline tool for describing and configuring VOLCANO and LIN networks.	www.volcanoautomotive.com
VOLCANO Target Package (VTP): > Configuration Tool Generator (Vcfg)	> Uses network information and node-related information to configure the target code to communicate over a network.	
> Target Code	> Provides the application code with a signal API.	
> Bootloader for In-Vehicle Software Download	> Bootloader to download software over the CAN-bus for end-of-line programming.	
PHYTEC		
MPC555 (phyCORE)	32-bit power (using PowerPC ISA) with dual full 2.0B TouCAN in credit card-sized package. Provides rapid development with MPC555 in a cost-effective, high-performance, single-board computer.	www.phytec.com
CANopen Slave	The CANopen Slave software is a network protocol for the development of devices according to the CANopen standard.	
CANopen Master	The CANopen Master software is intended for the development of network nodes with master functionality.	
Dearborn Group Technology		
Dearborn Protocol Adapters (DPA II+, DPAIII):	PC-to-Automotive Communication Networks Gateway	www.dgtech.com
Dearborn Programmable Bridge (DPB)	Custom software that translates and exchanges messages between various protocol networks.	
Dearborn Protocol Snooper	Hand-held device to monitor messages on various protocol networks.	
Falcon Flight Recorder	Data Capture and Manipulation for Gryphon Hardware	
Gryphon	Hardware interface that provides remote connectivity through an Ethernet connection.	
Hercules	Windows CAN Analyzer Software for Gryphon	
Network Analyzer Software (NAS)	Software to Troubleshoot Various Protocol Networks	
Super CAN Analysis Tool (S-CAT)	Hardware and software that supports advanced communication with in-vehicle networks.	

Disclaimer

This document may not include all the details necessary to completely develop this design. It is provided as a reference only and is intended to demonstrate the variety of applications for the device.

Related Documentation[Note]		
Document Number	**Description**	**Additional Information**
ADPAK	Analog ICs Integrated Solutions Pitch Pak	www.freescale.com
AN1776	Stereo Audio Transmission with *TouCAN*™	
AN1798	CAN Bit Timing Requirements	
AN1828	Flash Programming via CAN	
AN2010	Using The Freescale Semiconductor MSCAN Filter Configuration Tool	
AN2011	The MSCAN on the MCS912DP256 Versus HC12 Family	
AN2255	MSCAN Low-Power Applications	
AN2283	Freescale Semiconductor Scalable Controller Area Network (MSCAN) Interrupts	
EB376	A comparison of the MC9S12DP256 (mask set 0K36N) versus the HC12	
SG187	Automotive Selector Guide	
SG1002	Analog Selector Guide	
Note: Search on the listed document number.		

Notes

Learn More: Contact the Technical Information Center at +1-800-521-6274 or +1-480-768-2130.
For more information about Freescale products, please visit **www.freescale.com**.

SG2032
REV 3
6/2005