



Behind each and every benefit realized in a new control system is a significant software engineering investment designed to deliver increased productivity through improved process control and information flow. By choosing DeviceNet™ products from ODVA's member companies, users can protect this investment in several ways.

First, DeviceNet is a proven, stable network technology designed to meet the performance and reliability requirements of the industrial environment. DeviceNet uses CAN (Controller Area Network) for its data link layer, and CIP™ (Common Industrial Protocol) for the upper-layers of the network. DeviceNet is an open standard managed by ODVA and accepted by international standards bodies around the world. In addition, users will appreciate the seamless bridging and routing provided by DeviceNet to other CIP-based networks which currently include EtherNet/IP™ and ControlNet™.

Second, DeviceNet is supported by vendors around the world. Over 700 Vendor IDs have been issued by ODVA. The fact that so many vendors have chosen to implement DeviceNet in their products allows users to employ best-in-class products from vendors around the world who are best suited to support them based on application expertise and geographic coverage.

Third, DeviceNet CONFORMANCE TESTED® products have been certified by ODVA to conform to the specification and operate in open, multi-vendor systems by having passed conformance testing at one of ODVA's authorized conformance test service providers and having received an official Declaration of Conformity from ODVA.

Here's a look at the technology behind every DeviceNet product.

Introduction

DeviceNet is a digital, multi-drop network that connects and serves as a communication network between industrial controllers and I/O devices. Each device and/or controller is a node on the network. DeviceNet is a producer-consumer network that supports multiple communication hierarchies and message

prioritization. DeviceNet systems can be configured to operate in a master-slave or a distributed control architecture using peer-to-peer communication. DeviceNet systems offer a single point of connection for configuration and control by supporting both I/O and explicit messaging. DeviceNet also has the unique feature of having power on the network. This allows devices with limited power requirements to be powered directly from the network, reducing connection points and physical size.

DeviceNet follows the Open Systems Interconnection (OSI) model, an ISO standard for

Network Size	Up to 64 nodes
Network Length	Selectable end-to-end network distance varies with speed 125 Kbps 500 m (1,640 ft) 250 Kbps 250 m (820 ft) 500 Kbps 100 m (328 ft)
Data Packets	0-8 bytes
Bus Topology	Linear (trunkline/dropline); power and signal on the same network cable
Bus Addressing	Peer-to-Peer with Multi-Cast (one-to-many); Multi-Master and Master/Slave special case; polled or change-of-state (exception-based)
System Features	Removal and replacement of devices from the network under power

Table 1: Summary of DeviceNet Features and Functionality

network communications that is hierarchical in nature. Networks that follow this model define all necessary functions from the physical implementation up to the protocol and methodology to communicate control and information data within and across networks.

The Physical Layer

DeviceNet uses a trunk-line/drop-line topology that provides separate twisted pair busses for both signal and power distribution. The possible variants of this topology are shown in Figure 1. Thick or thin cable can be used for either trunklines or droplines. End-to-end network length varies with data rate and cable thickness as shown in Table 2.



DeviceNet™
Technical Overview

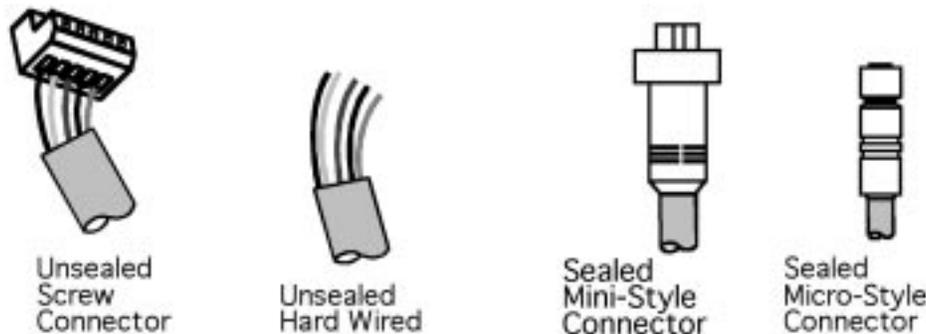


Figure 2: Open and Sealed Connectors are available on DeviceNet

The Data Link Layer

The Data Link Layer of DeviceNet is defined by the CAN specification and by the implementation of CAN Controller chips. The CAN specification defines two bus states called dominant (logic 0) and recessive (logic 1). Any transmitter can drive the bus to a dominant state. The bus can only be in the recessive state when no transmitter is in the dominant state. This fact comes into play in the bus arbitration scheme employed by CAN.

Several frame types are defined by CAN:

- data frame
- remote frame
- overload frame
- error frame

Data is moved on DeviceNet using the data frame. The other frames are either not used on DeviceNet or are for exception handling. The DeviceNet data frame format is shown in Figure 3.

CAN is a carrier sense network. Any node can attempt to transmit a message when no other nodes are transmitting. This feature provides inherent peer-to-peer capability. If two or more CAN nodes try to access the network simultaneously, a non-destructive, bit-wise arbitration mechanism resolves the potential conflict with no loss of data or bandwidth. All receivers on a CAN network synchronize to the transition from recessive to dominant represented by a Start of Frame bit. The identifier and the RTR (Remote Transmission Request – not used by DeviceNet) bit together form the Arbitration Field. The Arbitration Field is used to facilitate media access priority. When a device transmits, it also monitors (receives) each bit it sends to make sure

it is the same. This allows detection of simultaneous transmission. If a node transmitting a recessive bit receives a dominant bit while sending the arbitration field, it stops transmitting. The winner of arbitration between all nodes transmitting simultaneously is the one with the lowest numbered 11-bit identifier. CAN also specifies a data frame format with a 29-bit identifier field that is not used by DeviceNet.

The Control Field contains two fixed bits and a 4-bit length field. The length may be any number from 0 to 8 representing the number of bytes in the Data Field. The 0–8 byte size is ideal for low-end devices with small amounts of I/O data that must be exchanged frequently. With eight bytes, there is enough flexibility for simple devices to send diagnostic data, or to send a speed reference and acceleration rate to a drive.

DeviceNet also defines a fragmentation protocol that provides a way for nodes to transmit larger amounts of data with minimal protocol overhead.

The CRC field is a cyclic redundancy check word which is used by CAN controllers to detect frame errors. It is computed from the bits that come before it. A dominant bit in the ACK slot means at least one receiver besides the transmitter heard the transmission.

CAN provides very robust error checking and fault confinement employing several types of error detection and fault confinement methods including CRC and automatic retries. These methods, which are mostly transparent to the application, prevent a faulty node from disrupting the network.



DeviceNet™

Technical Overview

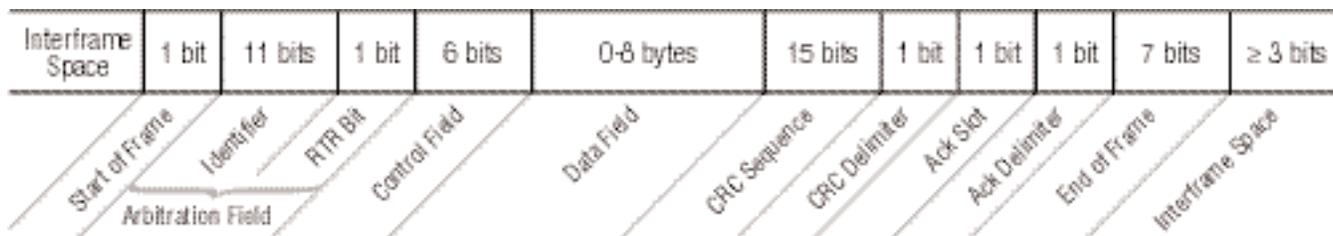


Figure 3. CAN Data Frame

The Network and Transport Layers

DeviceNet requires that a connection with a device must first be established in order to exchange information with that device. To establish a connection, each DeviceNet product will implement either an Unconnected Message Manager (UCMM) or a Group 2 Unconnected Port. Both perform their function by reserving some of the available CAN identifiers. When either the UCMM or the Group 2 Unconnected Port is used to establish an Explicit Messaging Connection, that connection is then used to move information from one node to the other, or to establish additional I/O Connections. Once I/O connections have been established, I/O data may be moved among devices on the network. At this point, all the protocol of the DeviceNet I/O message is contained within the 11-bit CAN identifier. Everything else is data.

The 11-bit CAN identifier is used to define the connection ID. Uniqueness of connection IDs is strictly controlled, which is required to take full advantage of producer-consumer capabilities. DeviceNet divides the 11-bit CAN identifier into four groups. The first three defined groups contain two fields; one 6-bit field for MAC ID and the other for Message ID. The combined fields define the connection ID. Four messages are used for offline communications.

Devices may be Clients or Servers or both. Clients and Servers may be producers, consumers or both. In a typical Client device, its connection would produce requests and consume responses. In a typical Server device, its connections would consume requests and produce responses. DeviceNet provides for several variations on this model. Some connections in either a Client or a Server may only consume messages. These connections

would be the destination for Cyclic or Change-of-State messages. Similarly, some connections in either a Client or Server may only produce messages. These connections would be the source for Cyclic or Changes-of-State messages.

The use of Cyclic and Change-of-State connections can substantially reduce bandwidth requirements. By design, nodes in a DeviceNet system are responsible for managing their own identifiers. These identifiers are interleaved (distributed) throughout the entire range. All nodes have a full range of message priorities available to them regardless of their Media Access Code Identifier (MAC ID). Through the duplicate MAC ID algorithm, the uniqueness of CAN identifiers is guaranteed without the need for a central tool or record for each network.

Figure 4 shows the DeviceNet allocations within the 11-bit CAN Identifier.

A related issue is detection of duplicate nodes. Because DeviceNet uses a device address inside the CAN Identifier Field, it presents a mechanism for detecting nodes with duplicate addresses. Preventing duplicate addresses is better than trying to locate them after they occur — something not taken into account in other CAN-based networks. Another key benefit to nodes managing their identifiers is that a user can add and delete nodes and/or add additional peer-to-peer messages among existing nodes at any time without having knowledge of the existing set-up. No centralized record must be located or reconstructed. Since nodes know which IDs are already in use, a tool simply has to request an I/O connection be added between the two devices, specifying priority level, the data path (class, instance, attribute) and the production trigger (cyclic, poll, or change-of-state).



DeviceNet™

Technical Overview

IDENTIFIER BITS											HEX RANGE	IDENTITY USAGE			
10	9	8	7	6	5	4	3	2	1	0					
0	Group 1 Message ID				Source MAC ID							000 - 3f	Message Group 1		
1	0	MAC ID				Group 2 Message ID							400 - 5f	Message Group 2	
1	1	Group 3 Message ID			Source MAC ID							600 - 7bf	Message Group 3		
1	1	1	1	1	Group 4 Message ID (0 - 2f)									7e0 - 7ef	Message Group 4
1	1	1	1	1	1	1	X	X	X	X				7f0 - 7f	Invalid CAN Identifiers
10	9	8	7	6	5	4	3	2	1	0					

Figure 4. DeviceNet Allocations of the 11-Bit CAN identifier field

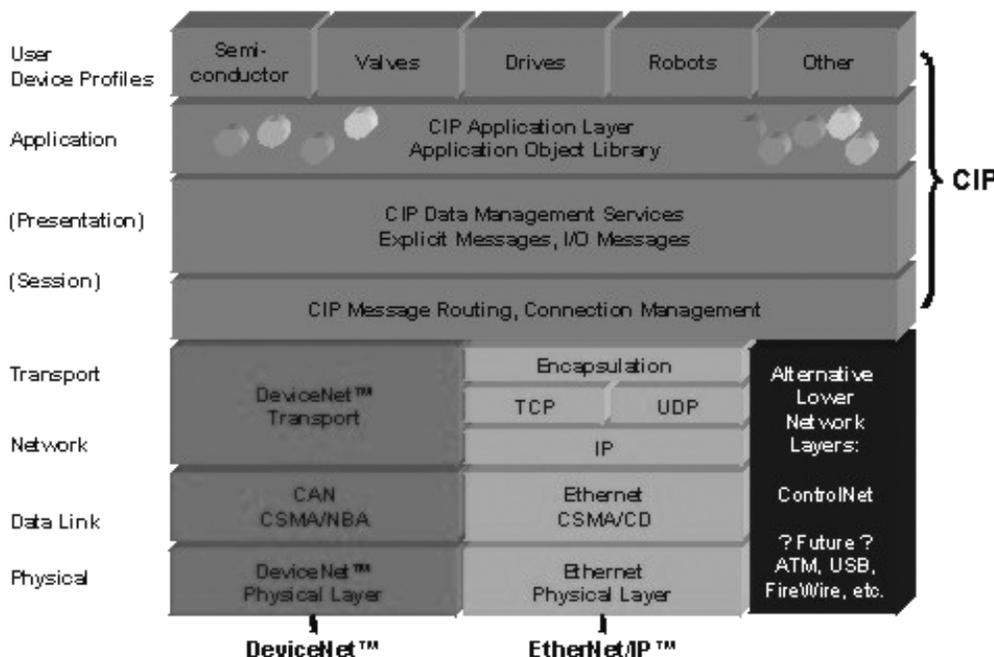
The Upper Layers: the Common Industrial Protocol (CIP)

DeviceNet uses the Common Industrial Protocol, called CIP, for its upper-layers. CIP is strictly object oriented. Each object has attributes (data) and services (commands) and behavior (reaction to events). Two different types of objects are defined in the CIP specification: communication objects and application-specific objects. Vendor-specific objects can also be defined by product makers for situations where a product requires functionality that is not in the specification.

For a given device type, a minimum set of common objects will be implemented. The user benefits from interoperability among devices regardless of the manufacturer or the device type.

Since CIP-based networks are based on a common application layer, the application data remains the same regardless of which network hosts the device. The application programmer doesn't even need to know to which network a device is connected. Figure 5 provides an overview of CIP within the context of DeviceNet and the OSI Model.

Figure 5: The CIP Object Model





DeviceNet™

Technical Overview

CIP also defines device profiles, which identifies the minimum set of objects, configuration options and the I/O data formats for different types of devices. Devices that follow one of the standard profiles will have the same I/O data and configuration options, will respond to all the same commands and will have the same behavior as other devices that follow that same profile. A subset of the device profiles supported by CIP and DeviceNet are shown in Table 3.

Device Profile	Device Type No.
AC Drives	02 _{hex}
Communications Adapter	0C _{hex}
Contactors	15 _{hex}
DC Drives	13 _{hex}
DC Power Generator	1F _{hex}
General Purpose Discrete I/O	07 _{hex}
Generic Device	00 _{hex}
Human-Machine Interface	18 _{hex}
Inductive Proximity Switch	05 _{hex}
Limit Switch	04 _{hex}
Mass Flow Controller	1A _{hex}
Motor Overload	03 _{hex}
Motor Starter	16 _{hex}
Photoelectric Sensor	06 _{hex}
Pneumatic Valve(s)	1B _{hex}
Position Controller	10 _{hex}
Process Control Valve	1D _{hex}
Residual Gas Analyzer	1E _{hex}
Resolver	09 _{hex}

Table 3: Device Types Supported by DeviceNet and CIP

DeviceNet also has standard mechanisms for vendors to incorporate their own features (objects, attributes, services beyond those defined in the device profile) into their products in addition to the minimum required objects. However, these additional features must be implemented in strict accordance with the DeviceNet specification in the manner prescribed by the specification.

Another important feature that sets CIP-based networks apart from other network architectures is the ability to originate a message on one CIP-based network such as DeviceNet, and then pass it to another CIP-based network such as EtherNet/IP with no presentation at the application layer. This seamless bridging and routing capability sets DeviceNet apart from other field bus networks. It means that a set of objects included in the specification define the mechanisms that a bridging device can use to forward the contents of a message from one network port to another without acting on the contents of that message. When using bridging devices that support these objects, the user's only responsibility is to describe the path that a message must follow. CIP ensures that the message is handled correctly, independent of the networks involved.

CIP is a producer/consumer-based model, rather than a traditional source/destination model. Producer-Consumer networks provide for more efficient use of networking bandwidth. A message is produced onto the network, it is identified, not by its destination address, but by connection ID. Multiple nodes may then consume the data to which this connection ID refers. The result of this dynamic connection approach provides two clear benefits in the efficiency of the network:

- If a node wants to receive data, it only needs to ask for it once to consume the data each time it is produced.
- If a second (third, fourth, etc.) node wants the same data, all it needs to know is the connection ID to receive the same data simultaneously with all other nodes.



The Predefined Master/Slave Connection Set

Fundamentally, DeviceNet employs a peer-to-peer messaging model. However, it also provides for a simplified communication scheme based on a Master/Slave relationship. This predefined connection scheme is known as the Predefined Master/Slave Connection Set. This connection method simplifies the movement of the I/O messages most often used in control applications.

Many sensors and actuators are designed to perform some predetermined function in which the type and amount of data the device will produce and/or consume is known at power-up.

The Predefined Master/Slave Connection Set provides connection objects that are almost entirely configured at the time the device powers-up. After powering up the network, the only remaining step necessary to begin the flow of data is for a “master” device to claim ownership of this predefined connection set within its “slave(s).” Slave devices can produce data using one or more of the following message types, depending on how the device is configured and the requirements of the application, as shown in Table 4.

Type	Description of Operation
polled	A slave configured for polled I/O will receive “output” data from the master device in a sequential order that is defined by the master’s scan list. The master’s polling rate is determined by the number of nodes in the scan list, the DeviceNet baud rate, the size of messages produced by the master and each node in its scan list and the internal timing of the master device. For a given system configuration, this messaging method will provide deterministic behavior. Polled I/O “output” data can be unicast or multicast.
cyclic	A device configured to produce a cyclic I/O message will produce its data at a precisely defined interval. This type of I/O messaging allows the user to configure the system to produce data at a rate appropriate for the application. Depending on the application this can reduce the amount of traffic on the wire and more efficiently use the available bandwidth.
change-of-state	A device configured to produce change-of-state (COS) message will produce data whenever it changes, or at a base heartbeat rate. This adjustable heartbeat rate provides a way for the consuming device to know that the producer is still alive and active. DeviceNet also defines a user-configurable Production Inhibit Time that limits how often COS messages are produced to prevent nodes from “flooding” the bandwidth. Users can adjust these parameters to provide optimum bandwidth utilization in a given application scenario.

Table 4: Slave I/O Message Types in the Predefined Master/Slave Connection Set



Conformance Testing

ODVA Conformance Testing is designed to validate that a product complies with relevant aspects of the ODVA specification, from the physical layer through the application layer. Devices that have passed ODVA DeviceNet conformance testing have passed all of the following tests at one of ODVA's authorized test service providers:

- ✓ **Control and Information Protocol (CIP) test** verifies that a product meets requirements of CIP and the additional messaging and services required for DeviceNet. It tests conformance at the device profile, application layers.
- ✓ **DeviceNet Electronic Data Sheet (EDS) test** verifies that the product's EDS (an ASCII text file supplied by the manufacturer) contains the correct grammar and all the required product information, including the device's identity, supported message types, configuration parameters and parameter enumeration. This file is used by network configuration software tools to help set up devices on the network.
- ✓ **DeviceNet Physical Layer test** verifies that a DeviceNet product's physical layer is designed and operating correctly. For example, it confirms that voltage and current levels remain in allowed ranges during operation and that the connectors and indicators meet ODVA specifications.

- ✓ **DeviceNet™ System test** verifies that the product operates in a multi-vendor DeviceNet™ system. The product is installed in a 64-node, multi-vendor environment and run in a variety of situations specifically designed to identify potential interoperability and system problems. Products must successfully establish explicit and I/O connections during various power-up sequences, and they must survive the on-line removal and replacement of other nodes. The test also confirms that the device will operate under heavy network traffic with scanners from multiple manufacturers and that the EDS file can be used to configure the device.

ODVA also offers specialized testing:

- ✓ **DeviceNet Power Supply test** verifies that network power supplies provide the necessary voltage and currents, and operate as the specification requires, under a variety of loading conditions.
- ✓ **Semiconductor test** is another level of testing for DeviceNet products used on semiconductor manufacturing tools. Products submitted for semiconductor-specific tests must first pass the standard test suite. This additional level of testing verifies conformance of connectors, indicators, switches, isolation, power, and object behavior with the ODVA specification supplement Interface Guidelines for DeviceNet Devices on Semiconductor Manufacturing Tools, including compliance with CE and SEMI S2.

Once a product passes the appropriate conformance tests, ODVA grants the vendor the right to use its conformance marks. A list <www.odva.org/10_2/04_products/04_prodtest.htm> of conformance-passed products also appears on the ODVA web site. Users who wish to specify only products that have passed conformance tests can find them listed online or identify them by these product marks.

www.odva.org
odva@odva.org



DeviceNet™ and CIP™ are trademarks of ODVA. DeviceNet CONFORMANCE TESTED® is a registered certification mark of ODVA. EtherNet/IP™ is a trademark used under license by ODVA. ControlNet™ is a trademark of ControlNet International.