K.W. Young, R.T. McLaughlin & S.B. Khoh[1]
University of Warwick[2]

DeviceNet Interoperability and Compliance

Abstract

DeviceNet has emerged as one of the low level fieldbusses optimised for industrial control. It uses the robust and powerful CAN (Controller Area Network) technology as the backbone. Interoperability between various DeviceNet devices, advanced failure prevention and fault diagnosis, and lower implementation costs are some of the immediate advantages of DeviceNet. Interoperability brings the issue of compatibility and conformance among the DeviceNet devices from various vendors. This paper describes the issues involved in implementing a DeviceNet system both from developers' and end-users' point-of-views. It investigates the 'plug-and-play' and interoperability of DeviceNet devices. A study for realising a fully automated compliance test for DeviceNet is done.

1.0 Introduction

The application of discrete fieldbus[3] technology for communication between factory instruments and devices have been on the increase in recent years. These fieldbusses namely Profibus, Fip, Lonworks, DeviceNet, SDS(Smart Distributed Devices), etc. may change the present scene of factory automation and control. They offer the users 'plug-and-play' interoperability, reconfiguration flexibility, advanced failure prevention and fault diagnosis, lower implementation costs as well as shorter commissioning time. (An Italian power utility predicts a 4% reduction in overall investment costs after allowing for a 10-20 % increase in device costs [3]). To date, there is no one fieldbus that caters for all of the manufacturing world's needs. The scenario consists of various fieldbus standards working in concert to achieve this new era in production automation control. DeviceNet was introduced as one of the many low level fieldbusses optimised for real-time control in industrial applications. This paper gives a brief overview of DeviceNet and describes the issues involved in implementing a DeviceNet system both from developers' and end-users' point-of-view. It discusses the 'plug-and-play' and interoperability issues, and highlights on the compliance test of DeviceNet devices. In addition, a study into the automated approach for realising an automated DeviceNet compliance test is done.

2.0 Overview of DeviceNet

DeviceNet is one of the low level open standard fieldbusses suitable for real-time control in industrial applications. It uses the proven Controller Area Network (CAN) technology as a backbone. CAN has been an ISO standard (ISO 11898 and ISO 11519-2) since 1993. Originally designed for automotive applications by Bosch, CAN is a robust network suitable for harsh environment operation. Its high performance error detection mechanism and good electromagnetic immunity make DeviceNet feasible to be used safely in the noisy factory environment. In addition, the use of CAN technology in production cars, and hence the high volume production of the silicon, make DeviceNet implementations more cost effective. This allows DeviceNet network interfaces to be economically implemented on simple, low cost field devices such as proximity switches.

[1] Email:young_k@eeyore.wmg.warwick.ac.uk, mclaughlin_r@eeyore.wmg.warwick.ac.uk, s.b.khoh@warwick.ac.uk
[2] Web site http://www.csv.warwick.ac.uk/
[3] Fieldbus is a low level industrial computer networks optimised for real-time information exchange and control. It implements only 3 layers of the ISO 7498 OSI model (ie. Physical, Data Link & Application).

Figure 1 shows how DeviceNet maps across the ISO 7498 and ISO 11898 (which is CAN). DeviceNet effectively specifies the application layer of the OSI model and media *layer (Layer 0, which is not defined in the ISO 7498-Basic Reference Model*). The application layer of the DeviceNet model standardises the common messaging mechanism so that information can be exchanged across the whole spectrum of DeviceNet devices. DeviceNet specific cables and connectors offer 'plug-and-play' compatibility among devices. A DeviceNet network supports 64 physical nodes on one of three different baud rates, i.e. 125kbit/s, 250 kbit/s and 500 kbit/s. It is configured using the bus network topology with CAN's CSMA/CD+NDBA (Carrier Sense Multiple Access/Collision Detection with Non-Destructive Bitwise Arbitration) bus access method for guaranteed data delivery. DeviceNet uses (Non-Return-to-Zero)NRZ baseband encoding technique for transmission over a shielded twisted pair cable. As with any serial communications, it has distance to speed limitations. At 125kbit/s, the DeviceNet network can be extended over a distance of 500 metres without any network repeater. However, the distance falls to only 100 metres if 500kbit/s is selected. Each of the 64 physical nodes are identified by a unique MAC (Media Access Control) identifier (i.e. MAC ID 0 to MAC ID 63) [1]. In order to prevent two nodes from having the same address (or MAC ID), every DeviceNet device must perform the duplicate MAC ID detection test upon power-up.

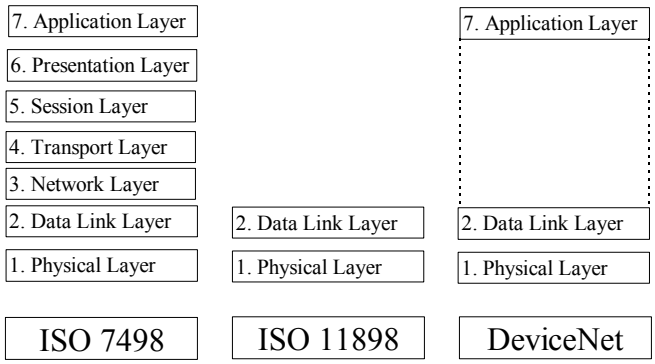

Figure 1 The relationship between ISO 7498, ISO 11898 and DeviceNet

2.1 Object Oriented Approach of DeviceNet

As mentioned by [4], DeviceNet is abstractly modelled as a collection of DeviceNet objects. Briefly, each DeviceNet device will have a communication object and an application object. The communication object will be responsibled for implementing the DeviceNet protocol messaging. Conversely, the application object is mainly concerned with the implementation of product specific features, e.g. the on/off state of a proximity switch. Figure 2 shows an example of the abstract model. The application object of the photosensor (MACID #63) contains the discrete states (on/off) of the sensor and the corresponding configuration (e.g. dark/light sensing). On the other hand, the communication object on the same device contains all the necessary DeviceNet protocol messaging. The communication object can be further divided into the 4 minimum object classes. They are:-

1. Connection object
2. DeviceNet object
3. Identity object
4. Message router object

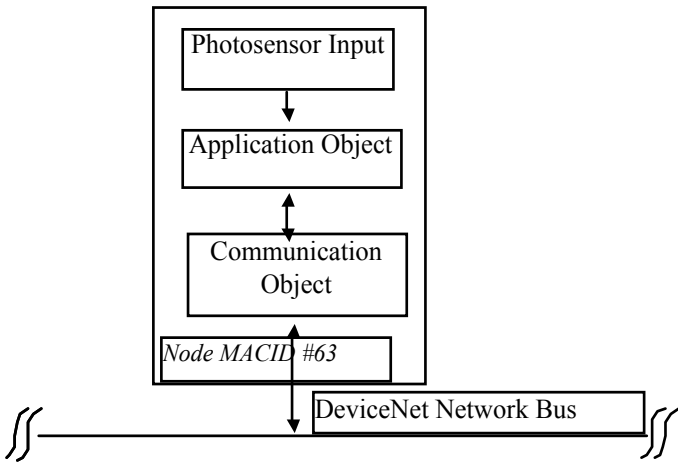

Figure 2 An example of DeviceNet abstract model

The connection object is responsible for implementing the logical connection between the producer and consumer of the data. In a connection based network such as DeviceNet, every device must have their services logically linked to the appropriate data producer or consumer. As such it is the responsibility of the Connection object to handle the open and close Explicit Message Connection(EMC) request/response, create the I/O message connection for cyclic, change-of-state and event driven data etc. The DeviceNet object performs the configuration and physical attachment such as MACID, baud rate, bus-off interrupt status and allocate/release the predefined master/slave connections. Data such as Vendor id, device type, serial number and other configuration data for device identification will be contained in the identity object. The identity object only serves its functions during initialisation and does not take part in the normal real-time communication. The message router object routes a service/response to the specified object/service source[1, 2].

2.2 DeviceNet Messaging

In general, DeviceNet messages can be divided into two different types, i.e. Explicit messages and I/O messages using the Message Groups 1, 2 and 3. Message Group 4 is reserved for future use as depicted in Figure 3. Due to the CSMA/CD+NDBA bus access method of CAN, Message Group 1 will have the highest priority to gain access to the bus (i.e. the lower the CAN identifier number, the higher the priority). Within the Message Group 2, the device which has the lower MAC ID will win the arbitration when bus contention occurs. In addition to the MAC ID, devices using Message Group 1 and Message Group 3 also arbitrate on the Message ID. For instance, Device A (MACID 2, Message ID 10) in Message Group 1 will loose arbitration to Device B (MACID 63, Message ID 1) of the same message group.

11-bit CAN Identifiers											Range in Hex	
10	9	8	7	6	5	4	3	2	1	0		
0	Group 1 Message ID				Source MAC ID						000-3FF	Message Group 1
1	0	MAC ID						Group 2 Message ID			400-5FF	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						7C0-7EF	Message Group 4
1	1	1	1	1	1	1	X	X	X	X	7F0-7FF	Invalid CAN Identifiers

highest priority

Figure 3 The arrangement of Message Groups of DeviceNet in CAN identifier field [1]

Based on CAN technology, a typical DeviceNet data packet ranges between 1 and 8 bytes, with an 11 bit address header. DeviceNet is designed to carry frequent short messages. The network

efficiency[4] of a DeviceNet network varies between 15.38% for 1 byte of I/O data to 59.26% for 8 bytes of I/O data (without stuff bits). A higher level field bus such as FIP achieves 79.63 % efficiency when transferring 128 bytes of data per packet, but the figure drops to 2.96% (based on turnaround time of 70μs) when 1 data byte is transferred[7]. The shorter data packets of DeviceNet improve the network access latency, i.e. the amount of time a node has to wait due to bus being busy before gaining bus access. Therefore, DeviceNet is very efficient when performing I/O messaging up to 8 data bytes. I/O data of more than 8 bytes will be transferred through Fragmentation Protocol. I/O Fragmentation Protocol uses the first byte of the CAN data field as protocol information, thus reducing the user data to 7 bytes per data packet. Similarly, Explicit messages utilise part of the CAN data field for protocol information. For instance, the Open Explicit Messaging Connection Request message which normally takes place during system initialisation, uses 4 data bytes to represent the Explicit Message Header, service code, requested message body format and message group. During real-time operation mode, the Explicit messages only use a data byte as protocol information (two data bytes for Explicit Message Fragmentation Protocol).

3.0 Working together(Interoperability)

The open standard nature of DeviceNet allows devices to be obtained from various sources and put together to work harmoniously on the same DeviceNet bus. This interoperability of DeviceNet open network architecture offers the end users with freedom of choice of supplier without having to rely on a single source for their automation devices. Competition among device vendors will benefit the end-users with better features and more cost effective products. However, since different vendor's devices can co-exist on the same network, it is of paramount importance to ensure that all the devices' behaviours are predictable. The idea is to prevent any device from disrupting the network operation. This requires devices to undergo a DeviceNet compliance process.

To further safeguard and promote the DeviceNet system, the Open DeviceNet Vendors Association (ODVA) has been formed. The ODVA consists of DeviceNet developers and acts as the governing body in ensuring that DeviceNet devices operate as stipulated in the DeviceNet specification. It is also responsible for forming the working committee in defining new device profiles. Typically, the device profile defines how the device behaves, how the device configuration is made and how the configuration affects its behaviour. It also defines the data format with which the DeviceNet communication is done. For example, the DeviceNet I/O assembly data format for a photoelectric sensor has been defined in Figure 4.

Data Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Diagnostic	Output

Figure 4 The I/O assembly data attribute format of a DeviceNet photoelectric sensor[2]

Any device which fulfils the fundamental requirements of the DeviceNet Specification is said to be compliant with DeviceNet. Optional features are allowed but they must be set to a default at power-up such that the device's behaviour is identical to the basic device type profile. The device developer may then provide various methods of accessing the optional features. These may include the use of the electronic data sheet(EDS) and parameter objects or through the more traditional way of a printed paper data sheet. The network configuration tool will then use the provided information to configure the device with the desired feature.

4.0 The setting up of a DeviceNet Network

In order to investigate the implementation implications of DeviceNet a flexible assembly cell has been set up at the International Manufacturing Centre of University of Warwick. This cell consists of two robots and a series of conveyors carrying pallets with two different sets of components on them. The robots perform different assembly tasks depending on the state of the pallets. This system was

[4] Network efficiency refers to the ratio of the amount of the user data and the number of bits needed to transmit such information[6].

originally wired conventionally and has been modified gradually to operate over DeviceNet. There are many different ways in which the trunk and drops can be assembled ranging from the use of simple screw connectors and bare cable to the use of moulded mini and micro connectors to fixed length cables. The system built uses every type of connection to find the merits of each. In a real application it is envisaged that only one style of connection would be used so that parts could easily be exchanged and modified. The conclusions from this exercise are that the more expensive option in terms of components is far easier to install. The decision as to which is the most cost effective is therefore dependent on the actual design for any individual network. If it is envisaged that the network will be modified very often during its life the ease of modification offered by the simple plug connections will more than pay for the increased initial investment cost.

For the network described the overall length is less than 25m so there is no limitation to the choice of baud rate. However the default speed is used as this does not require a node commissioning tool to be used prior to the node being introduced onto the system. Availability of such a tool would greatly improve the ease with which devices can be introduced to the network.

4.1 Device Commissioning

As with any network application, every device on a network must be set-up appropriately in order to establish communication. Ideally we would like to have the 'plug-and-play' device which is intelligent enough to perform the necessary operations to establish the logical connection. Even though there are devices which are capable of 'tuning' themselves to the correct baud rate, a device which is capable of establishing a logical connection without human intervention is still a long way off. Therefore, the definition of 'plug-and-play' is perhaps misleading. In DeviceNet for instance, every device must be configured using a network configuration tool such as the DeviceNet Manager Software. This allows the node number and baud rate to be set as well as any other programmable parameters. Some of these tools can also be used to program the master devices in order to establish the logical connection between them and their slaves. For a simple proximity switch for example the master device (in this case a PLC 5 scanner card[5]) needs to know the node number of the switch, the type of data transfer to be used (polled or strobed) and the data mapping. The data mapping tells the scanner card how many bits of data to receive or transmit, where to get them from and where to put them. The program in the PLC then looks very similar to a traditional program.

While putting a DeviceNet system together is undoubtedly easier than using traditional methods it is true to say that the system is still as complex but the complexity is handled within the configuration tools and software rather than within the wiring. It is even more important that the system is well documented as there is no way of looking at a device and working out which bits in the PLC memory it is mapped to, without the configuration tool. It is however safe to say that, because most of the configuration tools are PC based, there is every chance that the systems can automatically create much of the documentation needed.

The system builder needs little knowledge of how Device Net works in order to produce a working system. Some simple rules such as ensuring that the trunk is terminated, that the drop lengths are all within spec, that the power supplies can provide for the consumption of the devices and that the overall length restriction is not exceeded are all that is required. This relies very heavily on all devices working to the standard. If devices are used that fail to obey the standard tracing the rogue device can be very difficult. It is therefore imperative that a rigorous compliance test procedure has been performed by each device vendor.

5.0 Interoperability and Compliance Test

5.1 Compliance Test

DeviceNet compliance is the most important issue concerning every DeviceNet developer. Every effort must be made by the product developers to ensure that the developed product is DeviceNet compliant before the product is delivered to the customer or end-user. All devices bearing the name

[5] A scanner card is a communication adapter for interfacing between the DeviceNet (CAN) devices and the PLC's processor.

DeviceNet must strictly follow the DeviceNet specification in order to achieve device interoperability in an open standard network. Conformance to the DeviceNet specification can only be achieved if the same test plans or procedures are followed. This may involve a central neutral body to conduct the conformance test by executing the DeviceNet protocol implementations. Conversely, the responsibility of conformance testing may be brought in-house to the product developers themselves by using a standard set of compliance test procedures or software. The second approach of compliance test will depend heavily on the loyalty and responsibility of every product developer in safeguarding the DeviceNet protocol. It is in the interests of the ODVA and every device developer to prevent the scenario where vendor X's device upsets the DeviceNet network operations. In general, the DeviceNet compliant process can be briefly divided into two areas, i.e.

- the physical hardware compliance, and
- the software for protocol messaging compliance.

The physical hardware compliance will concern the physical layer tests to ensure that the hardware and transceiver used conform to the standard specification. This test ensures that the device-under-test (DUT) is capable of driving the network bus through a distance of 500m at 125kbit/s with 64 nodes on the bus. Other instances may involve the propagation delay test for the transceivers and opto-isolators to ensure that their behaviours fall within the limits specified in the test specification. Details of the hardware test procedures and specifications can be found in [5]. In short, this test is hardware oriented and requires the use of highly accurate measurement equipment (scopes/analysers/generators etc.) and a test rig.

Having made sure that the hardware conforms to the physical signalling defined in DeviceNet, the device-under-test (DUT) will undergo the software compliance test for DeviceNet protocol messaging. This area mainly concerns the testing of the DeviceNet communication object's software in the DUT to ensure that the minimum 4 objects, i.e. Connection object, DeviceNet object, Identity object and Message router object are implemented properly. For example, every group 2 device must have the capability to support the DeviceNet Explicit Messaging Connection(EMC). Therefore DeviceNet developers must hold the responsibility to ensure that every DeviceNet device manufactured complies with this part of the DeviceNet Specification. An important aspect of the DeviceNet conformance test is that it is only concerned with the DeviceNet communication object of the product. Even though the application object is equally as important as the communication object, the test of a device's functionality is not within the scope of this compliance test. It is therefore the responsibility of the device developers to guarantee the functionality and performance of their devices.

5.2 Interoperability Test

Having gone through the DeviceNet conformance test, a multi-vendor environment can be set-up to further test the DeviceNet devices as an integrated system rather than an individual DeviceNet device. DeviceNet developers can test their prototypes for correct inter-device communications and functionality within a multi-vendor environment. The system test may give some benchmarking figures for the DUT from a DeviceNet system point of view, or even some hints on how to optimise the protocol implementation. For example, in order to achieve a specific device performance, the device may need to be configured at a low MACID number on a busy network.

5.3 Automated Compliance Test Process

A study into the automated compliance testing process of DeviceNet devices is currently being done at the University of Warwick. If the 'DeviceNet device' is equipped with the parameter object or parameter object stubs and EDS(Electronic Data Sheet)[1, 2], it is possible to establish the identity of the device under test (DUT) without human intervention. The Identity Object contains information such as vendor id, device type, serial number etc. This information, mapped with that from the EDS will allow the Compliance Test Engine (CTE) to know the identity of the DUT. The CTE may then be able to generate the appropriate test list depending on the identity and complexity of the DUT. Having determined which tests to perform on the DUT, the CTE will then execute the tests and collect the test results. The test results will then be analysed to determine whether the device under test conforms to DeviceNet protocol. If the device fails to conform, then the CTE will illustrate what has gone wrong and will attempt to propose a solution to the problem. The CTE when completed will be useful for both

DeviceNet developers and compliance testers. The aim of having the CTE is to provide an accurate compliance test with minimal human intervention. The CTE will eventually link up with measuring equipment to realise the fully automated compliance test system.

6.0 Conclusion

DeviceNet is an open standard fieldbus which has the potential to become a de facto standard due to its performance capability and low cost. The success or failure of DeviceNet lies in the integrity and security of the network. As such the issue of compliance and interoperability is of utmost importance for both the end-users and developers. DeviceNet developers must strictly adhere to the DeviceNet specification at all times for conformance to ensure interoperability between devices is achieved. Ultimately the fate of DeviceNet, as with all of the other fieldbus standards, will not depend on the technological issues. In order to become a de facto standard it needs to be widely used and this will depend not only on a large range of compliant products being made available but also on the system builders deciding to use the technology. In addition, ODVA which consists of DeviceNet vendors must play a vital role in promoting DeviceNet. Further development must also be done to enable DeviceNet to keep abreast with the fast changing pace of technology. Many opportunities can be realised using this technology including decentralised control. Perhaps in the future rather than using a PLC or similar to control simple slaves, the sensors and actuators will communicate directly to execute simple control routines.

Glossary of Terms

CAN	Controller Area Network
CSMA/CD+NDBA	Carrier Sense Multiple Access/Collision Detection with Non-Destructive Bitwise Arbitration
CTE	Compliance Test Engine
DUT	Device Under Test
EDS	Electronic Data Sheet - An ASCII file which contains the corresponding device parameters to be used by DeviceNet configuration tools during device configuration process.
EMC	Explicit Messaging Connection
ISO	International Standard Organisation[6]
MACID	Medium Access Control Identifier
NRZ	Non-Return-to-Zero
ODVA	Open DeviceNet Vendors Association[7]
OSI	Open Systems Interconnection

[6] Web site http://www.iso.ch/
[7] Web site http://www.industry.net/odva/

References

[1] "*DeviceNet Specification - Volume I, Release 1.1*", Allen-Bradley Publication no. 1787-6.5.1 August 1994

[2] "*DeviceNet Specification - Volume II, Release 1.0*", Allen-Bradley Publication no. 1787-7.1 August 1994

[3] "*Fieldbus - The Executive Guide*", Department of Trade and Industry Publication, 1993

[4] D.Noonen, S. Siegel, P. Maloney, "*DeviceNet Application Protocol*", Proceedings of the 1st. International CAN Conference 1994, Mainz-Germany, pp. 10:11-10:19.

[5] E. Polce, P. Maloney, "*DeviceNet Compliancy Test Plan : Test-500/002/01, Version 1.0*", Allen-Bradley, October 1994.

[6] G.Cena, L.Durante, A.Valenzano, "*Standard Fieldbus Networks for Industrial Applications*", Computer Standards & Interfaces 17 (1995) pp. 155-167

[7] Philippe Leterrier, "*The FIP Protocol*", Centre de Compétence FIP