Huron Net Works, Inc.

DN-400 Family

DeviceNet Serial Link Adapter I/O Modules

Device Profile

DN-410	Firmware Revision	3.6
DN-410-05	Firmware Revision	3.6
DN-400	Firmware Revision	3.1
DN-400-05	Firmware Revision	3.1
DN-TC400	Firmware Revision	3.2
DN-TC400-05	Firmware Revision	3.2

Publication #2200093 Revision History

Rev	Date	Note(s)
0.1	5/1/02	Original – Pre-Release
0.2	5/2/02	Change to Specification Template & Minor Corrections
0.3	5/17/02	Implementation Updates, Add Serial Link, UART, & Non-Volitile Test Objects
0.4	6/12/02	Update for prototype release on 6/6/02, add Parameter Object and Debug Object
0.5	6/28/02	Merge the three DN-400 family products into a unified specification
0.6	3/15/03	Define new products on half size boards
0.7	5/28/03	Change name to “DN-400 Family…”. Update revision and Identity Object Information. Delete diagnostic and Factory Test Object Information.
0.8	4/12/04	Add IN3 and IN4 to DN-TC400 and DN-TC400.0.5
0.9	8/24/04	Delete: PCA Object Delete: Debug Object Delete: Serial Link Diagnostic Action ‘3’ Add: Quick Connect Add: Vendor Specific Parameter Object
1.0	2/21/05	Correct: Revision attribute values in Identity Object Table Add: Table of Contents

Table of Contents

Preface

In the revision 0.6 release of this document, three new variants of the 400 family products were documented on half-size boards. They use the same firmware as their larger versions but from the network point of view they have new identities.

This document now includes material from two other Device Profiles for the family of products, which includes the DN-400, the DN-TC400, and the newest member of the family the DN-410. A single printed circuit board is used for all three products, and the same firmware, as of this release date, is used for all three products. In addition there are configuration parameters, which allow a product to transform itself into another member of the family.

This document will supersede and obsolete the following documents:

#2200087-A.0	DN-400 Device Profile
#2200081-A.1	DN-TC400 Device Profile

References

1. DeviceNet Specification
2. DN-SLIP Functional Specification #2200092

1 Object Model

The DN-410 family are general purpose DeviceNet Serial Link I/O modules. There are up to four general purpose outputs, up to four general purpose inputs, and two power sense inputs. Members of the family can be either Device Type 0 (0x00 – Generic), or Device Type 12 (0x0C – Communications Adapter). All devices in the family support the same object model with some differences in the number of instances for some classes.

1.1 Objects Present in Device

Object	Class Id	Number of Instances		
		DN-410 DN-410-05	**DN-400 DN-400-05**	**DN-TC400 DN-TC400-05**
Identity	0x01	1	1	1
Message Router*	0x02	1-DeviceNet 1-Serial Link	1-DeviceNet 1-Serial Link	1-DeviceNet 1-Serial Link
DeviceNet	0x03	1	1	1
Assembly	0x04	4	4	4
Connection	0x05	1-Explicit 1-Poll I/O 1-COS/Cyclic	1-Explicit 1-Poll I/O 1-COS/Cyclic	1-Explicit 1-Poll I/O 1-COS/Cyclic
Discrete Input Point	0x08	6	6	6
Discrete Output Point	0x09	4	4	4
Parameter Object	0x0F	4	4	4
Acknowledge Handler Object	0x2B	1	1	1
Power Sense	0x64	1	1	1
Serial Link Object	0x6A	1	1	1
UART Object	**	1	1	1
Non-Volitile Test Object	**	1	1	1
Vendor Specific Parameter	**	1	1	1
Factory Test Object	**	1	1	1

* Message Router attributes and services are optional. Message Router behavior is required.
** These objects are described in the Vendor Specific Object Library

1.2 Objects That Effect Behavior

Object	Class Id	Effect on Behavior
Identity	0x01	Supports the reset service
Message Router	0x02	No effect
DeviceNet	0x03	Configures port attributes
Assembly #100	0x04	Input Data Format – from Serial Link
Assembly #101	0x04	Input Data Format – from Discrete Input Points
Assembly #120	0x04	Output Data Format – to Serial Link
Assembly #121	0x04	Output Data Format – to Discrete Output Points
Connection	0x05	Establishes the number of connections
Discrete Input Point	0x08	Reports state of position sensing inputs and Aux supply monitors
Discrete Output Point	0x09	Defines behavior of solenoid output points
Parameter Object	0x0F	Static Assembly/IO Selection
Acknowledge Handler Object	0x2B	Acknowledge Handling for COS/Cyclic IO Connections
Power Sense Object	0x64	Invert Value of Power Sense Inputs
Serial Link	0x6A	Configures and Defines Serial Link Attributes
UART Object	**	Access UART Registers
Non-Volitile Test Object	**	Test Non-Volitile Memory
Vendor Specific Parameter Object	**	Vendor Specific Configuration
Factory Test Object	**	Vendor Specific Configuration

** These objects are described in the Vendor Specific Object Library

1.3 Object Interfaces

Object	Class Id	Interface
Identity	0x01	Message Router
Message Router	0x02	Explicit Message Connection Instance
DeviceNet	0x03	Message Router
Assembly	0x04	I/O Connection or Message Router
Connection	0x05	Message Router
Discrete Input Point	0x08	I/O Connection or Message Router
Discrete Output Point	0x09	I/O Connection or Message Router
Parameter Object	0x0F	Message Router
Acknowledge Handler Object	0x2B	Message Router
Power Sense Object	0x64	Message Router
Serial Link	0x6A	Message Router
UART Object	**	Message Router
Non-Volitile Test Object	**	Message Router
Vendor Specific Parameter Object	**	Message Router
Factory Test Object	**	Message Router

** These objects are described in the Vendor Specific Object Library

2 Standard Objects

2.1 Identity Object (0x01)

There is a single instance of the identity object for the device.

2.1.1 Class Attributes

No class attributes or class services are supported.

2.1.2 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Attribute Values		
				DN-410 DN-410-05	DN-400 DN-400-05	DN-TC400 DN-TC400-05
1	Get	Vendor	UINT	0x0014	0x0014	0x0014
2	Get	Device Type	UINT	0x000C	0x0000	0x0000
3	Get	Product Code	UINT	0x0016 0x001B	0x0014 0x001A	0x0012 0x0019
4	Get	Revision	STRUCT	03.06	03.01	03.02
5	Get	Status	WORD	0x0001	0x0001	0x0001
6	Get	Base Serial #	UDINT	0x00005400 0x00006800	0x00004C00 0x00006400	0x00004000 0x00006000
7	Get	Product Name	SHORT_ STRING	6,DN-410 9,DN-410 SF	6,DN-400 9,DN-400 SF	16,Tool Changer I/O 19,Tool Changer I/O SF
8	Get	State	USINT	[0..5]	[0..5]	[0..5]
9	Get	CCV	UINT	[0..65535]	[0..65535]	[0..65535]
10	Get/Set	Heartbeat	USINT	[0.255]	[0.255]	[0.255]

2.1.3 Instance Services

Service Code	Service Name	Description of Service
05	Reset	Reset the device to power up configuration
0E	Get_Attribute_Single	Returns the contents of the specified Attribute
10	Set_Attribute_Single	Sets the contents of the specified attribute

2.1.4 Reset Service

Data Type	Value	Description of Service
UINT	0	Reset the device to power up configuration
	1	Return the device to Out-of-Box configuration

2.2 Message Router Objects (0x02)

There is a single instance of the Message Router Object. For this single instance of the Message Router Object there is no externally visible interface. There are no class attributes or services, and there are no instance attributes or services. This is consistent with the definition of the attributes and the services as optional. The behavior of this object is required.

2.3 DeviceNet Object (0x03)

There is a single instance of the DeviceNet Object for the device

2.3.1 Class Attributes

Attribute ID	Access Rule	Name	Data Type	Value
1	Get	Revision	UINT	0x0002

2.3.2 Class Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute

2.3.3 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Value
1	Get/Set	MACID Value	USINT	[0..63]
2	Get/Set	Baud Rate Value	USINT	[0..2]
3	Get/Set	BOI	BOOL	0x00 =Fault
4	Get/Set	Bus-Off Counter	USINT	0x00
5	Get	Allocation Information	STRUCT	Allocated Cnxns
6	Get	MAC ID Switch Changed	BOOL	0=No Change 1=Changed
7	Get	Baudrate Switch Changed	BOOL	0=No change 1=Changed
8	Get	MAC ID Switch Setting	USINT	[0..63]
9	Get	Baudrate Switch Setting	USINT	[0..3]
10	Get/Set	Quick Connect	USINT	0 = Disabled 1 = Enabled

2.3.4 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Sets the contents of the specified attribute
4B	Allocate	Creates predefined M/S connections
4C	Release	Deletes predefined M/S connections

2.4 Assembly Object (0x04)

There are four static assembly instances for each device, with instance numbers in the vendor specific range.

2.4.1 Class Attributes and Services

No class attributes or services are supported.

2.4.2 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Value
3	Get/Set	Value	array of bytes	see definitions

2.4.3 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Sets the contents of the specified attribute

2.4.4 Identification of I/O Assembly Instances

The assembly instances all come from the Vendor Specific Range, since the Device Profile for the Communications Adapter and the Generic Devices do not specify any assembly instances in the Open range.

Number	Static Assembly Type	Name
100	Input	Serial Link Inputs
101	Input	Hardware Inputs
120	Output	Serial Link Outputs
121	Output	Hardware Outputs

2.4.5 Format of I/O Assembly Data Attribute

2.4.5.1 Assembly #100 – Serial Link Inputs

The Serial Link Input Assembly contains from zero(0) to eight(8) bytes. This size is determined by a configurable parameter. The manufactured default size is four bytes. Parameter Class Instance 2 is used to control the size of the Serial Link Input Assembly

Byte\Bit	7	6	5	4	3	2	1	0
0	B7	B6	B5	B4	B3	B2	B1	B0
1	B15	B14	B13	B12	B11	B10	B9	B8
2	B23	B22	B21	B20	B19	B18	B17	B16
3	B31	B30	B29	B28	B27	B26	B25	B24
4	B39	B38	B37	B36	B35	B34	B33	B32
5	B47	B46	B45	B44	B43	B42	B41	B40
6	B55	B54	B53	B52	B51	B50	B49	B48
7	B63	B62	B61	B60	B59	B58	B57	B56

Bits B63..B0 are general purpose bits whose use and meaning is defined on the other end of the Serial Link. These Serial Link Inputs are consumed in a Poll Response Message on the Serial Link and atomically copied into an area of memory. These bits are then returned in a Poll Response Message on DeviceNet, when a Poll Request is received on DeviceNet. In the absence of valid data from the Serial Link, either a null frame or a configurable predefined message of the proper length is returned.

2.4.5.2 Assembly #101 - DN-410, DN-410-05, DN-400, DN-400-05 Hardware Inputs

Byte	7	6	5	4	3	2	1	0
0	PS2	PS1	0	0	IN4	IN3	IN2	IN1

IN1 J5.4
IN2 J5.2
IN3 J6.4
IN4 J6.2
PS1 0 = Vaux_sens > Vpower_supply
PS2 0 = Vaux > Vpower_supply

Vpower_supply is different on each device. It is nominally 20.4 volts with no specified tolerance. There is also hysteresis in the switching threshold of about 1/2 Volt. Negative logic on the Power Sense Inputs is the default behavior. This behavior can be changed to the opposite, or positive logic, behavior by the Power Sense Object (ClassId = 0x64).

Bit positions 4 & 5 are unused and will always be returned as zero. These bits can be returned in the Poll Response Message, or in the COS/Cyclic Message.

2.4.5.3 Mapping for Assembly Instance #101 – DN-410, DN-410-05, DN-400, DN-400-05 Hardware Inputs

Data Name	Class		Instance Number	Attribute	
	Name	Number		Name	Number
IN1	Discrete Input Point	0x08	1	Value	3
IN2	Discrete Input Point	0x08	2	Value	3
IN3	Discrete Input Point	0x08	3	Value	3
IN4	Discrete Input Point	0x08	4	Value	3
PS1	Discrete Input Point	0x08	6	Value	3
PS2	Discrete Input Point	0x08	7	Value	3

2.4.5.4 Assembly #101 – DN-TC400, DN-TC400-05 Hardware Inputs

Byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	0	0	IN4	IN3	PS2	PS1	IN2	IN1

IN1 J5.4
IN2 J5.2
IN3 J6.4
IN4 J6.2
PS1 0 = Vaux_sens > Vpower_supply
PS2 0 = Vaux > Vpower_supply

Vpower_supply is different on each device. It is nominally 20.4 volts with no specified tolerance. There is also hysteresis in the switching threshold of about 1/2 Volt. Negative logic on the Power Sense Inputs is the default behavior. This behavior can be changed to the opposite, or positive logic, behavior by the Power Sense Object (ClassId = 0x64).

Bit positions 7 and 6 are unused and will always be returned as zero. These bits can be returned in the Poll Response Message, or in the COS/Cyclic Message.

2.4.5.5 Mapping for Assembly Instance #101 – DN-TC400, DN-TC400-05 Hardware Inputs

Data Name	Class		Instance Number	Attribute	
	Name	Number		Name	Number
IN1	Discrete Input Point	0x08	1	Value	3
IN2	Discrete Input Point	0x08	2	Value	3
PS1	Discrete Input Point	0x08	3	Value	3
PS2	Discrete Input Point	0x08	4	Value	3
IN3	Discrete Input Point	0x08	6	Value	3
IN4	Discrete Input Point	0x08	7	Value	3

2.4.5.6 Assembly #120 – Serial Link Outputs

The Serial Link Output Assembly contains from zero(0) to eight(8) bytes. This size is determined by a configurable parameter. The manufactured default size is four bytes. Parameter Class Instance 3 is used to control the size of the Serial Link Output Assembly

Byte\Bit	7	6	5	4	3	2	1	0
0	B7	B6	B5	B4	B3	B2	B1	B0
1	B15	B14	B13	B12	B11	B10	B9	B8
2	B23	B22	B21	B20	B19	B18	B17	B16
3	B31	B30	B29	B28	B27	B26	B25	B24
4	B39	B38	B37	B36	B35	B34	B33	B32
5	B47	B46	B45	B44	B43	B42	B41	B40
6	B55	B54	B53	B52	B51	B50	B49	B48
7	B63	B62	B61	B60	B59	B58	B57	B56

Bits B63..B0 are general purpose bits whose use and meaning is defined on the other end of the Serial Link. These bits are normally consumed in a Poll Request Message, and atomically copied into an area of memory, which is used to construct a response for a request from the Serial Link.

2.4.5.7 Assembly #121 – DN-400, DN-400-05, DN-TC400, DN-TC400-05, DN-410, DN410-05 Hardware Outputs

Byte	7	6	5	4	3	2	1	0
0	Unused				OUT4	OUT3	OUT2	OUT1

Bit positions 4through 7 are unused and will be ignored by the device

2.4.5.8 Mapping for Assembly #121

Data Name	Class		Instance	Attribute	
	Name	Number	Number	Name	Number
OUT1	Discrete Output Point	0x09	1	Value	3
OUT2	Discrete Output Point	0x09	2	Value	3
OUT3	Discrete Output Point	0x09	3	Value	3
OUT4	Discrete Output Point	0x09	4	Value	3

2.5 Connection Object (0x05)

2.5.1 Class Attributes

No class attributes or services are supported

2.5.2 Instance Attributes

There are three instances of the Connection Object in the device. Instance #1 is assigned to the Explicit Message Connection. Instance #2 is assigned to the Polled I/O Connection and supports COS/Cyclic consumption and ack production. Instance #4 handles COS I/O production and ack consumption. The tables below show the attributes and the predefined values where applicable. In order to have defined numeric values in the tables below, a slave MAC Id of sixty-three(63), and a master MAC Id of one(1) are assumed.

2.5.3 Instance Services

Service Code	Service Name	Description of Service
05	Reset	Reset the connection - restart timer transition from timed out state.
0E	Get_Attribute_Single	Returns the contents of the specified Attribute
10	Set_Attribute_Single	Sets the contents of the specified Attribute

Explicit Message Connection (Instance #1)

Attribute ID	Access Rule	Name	Data Type	Value
1	Get	State	USINT	0x03
2	Get	instance_type	USINT	0x00
3	Get	Xport Class Trigger	USINT	0x83
4	Get	produced connection ID	UINT	0x5FB for MAC ID 63
5	Get	consumed connection ID	UINT	0x5FC for MAC ID 63
6	Get	initial comm characteristics	USINT	0x21
7	Get	produced connection size	UINT	0x0015
8	Get	Consumed connection size	UINT	0x0015
9	Get/Set	expected packet rate	UINT	Application Dependent
10	N/A	N/A	N/A	Obsolete, no longer used
11	N/A	N/A	N/A	Obsolete, no longer used
12	Get/Set	watchdog timeout action	USINT	0x01 Auto Delete
13	Get	produced path length	UINT	0x0000
14	Get	Produced path	Array of USINT	<NULL>
15	Get	consumed path length	UINT	0x0000
16	Get	consumed path	Array of USINT	<NULL>
17	Get	production inhibit timer	UINT	0x0000*

* Server connection endpoints do not use this timer

Poll/COS/Cyc I/O Message Connection (Instance #2)

Attribute ID	Access Rule	Name	Data Type	Value
1	Get	State	USINT	0x03
2	Get	instance_type	USINT	0x01
3	Get	Xport Class Trigger	USINT	0x82 0x80 no poll, no ack
4	Get	produced connection ID	UINT	0x3FF 0xFFFF no Ack for MAC ID 63
5	Get	consumed connection ID	UINT	0x5FD for MAC ID 63
6	Get	initial comm characteristics	USINT	0x01 0xF1 no poll, no ack
7	Get	produced connection size	UINT	See Configuration Table
8	Get	consumed connection size	UINT	See Configuration Table
9	Get/Set	expected packet rate	UINT	Application Dependent
10	N/A	N/A	N/A	Obsolete, no longer used
11	N/A	N/A	N/A	Obsolete, no longer used
12	Get/Set	watchdog timeout action	USINT	0x00 Time Out
13	Get	produced path length	UINT	0x0006 or 0x0000 no poll, no ack
14	Get	produced path	Array of USINT	See Configuration Table
15	Get	consumed path length	UINT	0x0006
16	Get	consumed path	Array of USINT	See Configuration Table
17	Get	production inhibit timer	UINT	0x0000 poll app dep. - COS/Cyc

COS/Cyc I/O Message Connection (Instance #4)

Attribute ID	Access Rule	Name	Data Type	Value
1	Get	state	USINT	0x03
2	Get	instance_type	USINT	0x01
3	Get	Xport Class trigger	USINT	0x12 COS 0x02 Cyc 0x10 COS, no Ack 0x00 Cyc, no Ack
4	Get	produced connection ID	UINT	0x37F for MAC ID 63
5	Get	consumed connection ID	UINT	0x5FA 0xFFFF no Ack for MAC ID 63
6	Get	initial comm characteristics	USINT	0x01 0x0F no Ack
7	Get	produced connection size	UINT	See Configuration Table
8	Get	consumed connection size	UINT	0x0000
9	Get/Set	expected packet rate	UINT	Application Dependent
10	N/A	N/A	N/A	Obsolete, no longer used
11	N/A	N/A	N/A	Obsolete, no longer used
12	Get/Set	watchdog timeout action	USINT	0x00 Time Out
13	Get	produced path length	UINT	0x0006
14	Get	produced path	Array of USINT	See Configuration Table
15	Get	consumed path length	UINT	0x0004 or 0x0000 no Ack
16	Get	Consumed path	Array of USINT	20.2B.24.01 <null> no Ack
17	Get/Set	production inhibit timer	UINT	app dependent

Connection Attribute Configuration Table

Attribute Name	Attribute Address	Configuration		
		DN-400 DN-400-05	DN-TC400 DN-TC400-05	DN-410 DN-410-05
Poll produced connection size	{5,2,7}	1 or [0..8]	1 or [0..8]	[0..8] or 1
Poll consumed connection size	{5,2,8}	1 or [0..8]	1 or [0..8]	[0..8] or 1
Poll produced path	{5,2,14}	…24.65… or …24.64… or <> no poll, no ack	…24.65… or …24.64… or <> no poll, no ack	…24.64…or …24.65… or <> no poll, no ack
Poll consumed path	{5,2,16}	…24.79… or …24.78…	…24.79…or …24.78…	…24.78… or …24.79…
COS/Cyclic produced connection size	{5,4,7}	1 or [0..8]	1 or [0..8]	[0..8] or 1
COS/Cyclic produced path	{5,4,14}	…24.65… or …24.64…	…24.65… …24.64…	…24.64… or …24.65…

2.6 Discrete Input(0x08)

2.6.1 Class Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Get	Revision	UINT	Revision of this object	2

2.6.2 Class Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute

2.6.3 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
3	Get	Value	BOOL	Input Point Value	0=off 1=on

2.6.4 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute

2.6.5 Input Point Mapping, DN-410, DN-410-05, DN-400, and DN-400-05

Data Name	Class		Instance Number	Attribute	
	Name	Number		Name	Number
IN1	Discrete Input Point	0x08	1	Value	3
IN2	Discrete Input Point	0x08	2	Value	3
IN3	Discrete Input Point	0x08	3	Value	3
IN4	Discrete Input Point	0x08	4	Value	3
PS1	Discrete Input Point	0x08	6	Value	3
PS2	Discrete Input Point	0x08	7	Value	3

2.6.6 Input Point Mapping, DN-TC400, DN-TC400-05

Data Name	Class		Instance Number	Attribute	
	Name	Number		Name	Number
IN1	Discrete Input Point	0x08	1	Value	3
IN2	Discrete Input Point	0x08	2	Value	3
PS1	Discrete Input Point	0x08	3	Value	3
PS2	Discrete Input Point	0x08	4	Value	3
IN3	Discrete Input Point	0x08	6	Value	3
IN4	Discrete Input Point	0x08	7	Value	3

2.7 Discrete Output (0x09)

2.7.1 Class Attributes

No class attributes or services are supported for this object.

2.7.2 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
3	Get/Set	Value	BOOL	Output Point Value	0=off 1=on
5	Get/Set	Fault Action	BOOL	Action taken on output's value in Recoverable Fault State	0=Fault Value 1=Hold Last State
6	Get/Set	Fault Value	BOOL	Value to use for Fault Action	0=off 1=on
7	Get/Set	Idle Action	BOOL	Action taken on output's value in Idle State	0=Idle Value 1=Hold Last State
8	Get/Set	Idle Value	BOOL	Value to use for Idle Action	0=off 1=on

2.7.3 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Modifies an attribute value

2.7.4 Output Point Mapping

Data Name	Class		Instance Number	Attribute	
	Name	Number		Name	Number
OUT1	Discrete Output Point	0x09	1	Value	3
OUT2	Discrete Output Point	0x09	2	Value	3
OUT3	Discrete Output Point	0x09	3	Value	3
OUT4	Discrete Output Point	0x09	4	Value	3

2.8 Parameter Object (0x0F)

The parameter object contains four parameter stubs. These parameters have no other mapping in the object model. They are used to select different I/O configurations, different I/O sizes, and configure the permission to override the soft settable baudrate and MacId of the device.

2.8.1 Class Attributes

Three class attributes are supported.

Attribute ID	Access Rule	Name	Data Type	Description	Value
2	Get	Max Instance	UINT	Maximum Instance Number	0x0004
8	Get	Parameter Class Descriptor	WORD	Bits which describe parameters	0x0009
9	Get	Configuration Assembly Instance	UINT	Instance number if used	0x0000 no configuration assembly

2.8.2 Class Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified Attribute

2.8.3 Instance #1 Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Set	Parameter Value	USINT	IO Configuration	0x00 = Serial Link IO 0x01 = Hardware I/O
2	Get	Link Path Size	USINT		0x00 No Link Path
3	Get	Link Path	ARRAY		<NULL> No Link Path
4	Get	Descriptor	WORD		0x0000
5	Get	Data Type	EPATH		0xC6
6	Get	Data Size	USINT		0x01

2.8.4 Instance #2 Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Set	Parameter Value	USINT	Serial Consumed Size	[0..8] Size of the Serial Link Input Array Default = 4
2	Get	Link Path Size	USINT		0x00 No Link Path
3	Get	Link Path	ARRAY		<NULL> No Link Path
4	Get	Descriptor	WORD		0x0000
5	Get	Data Type	EPATH		0xC6
6	Get	Data Size	USINT		0x01

2.8.5 Instance #3 Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Set	Parameter Value	USINT	Serial Produced Size	[0..8] Size of the Serial Link Output Array Default = 4
2	Get	Link Path Size	USINT		0x00 No Link Path
3	Get	Link Path	ARRAY		<NULL> No Link Path
4	Get	Descriptor	WORD		0x0000
5	Get	Data Type	EPATH		0xC6
6	Get	Data Size	USINT		0x01

2.8.6 Instance #4 Attributes

If the MacId and Baudrate are soft settable (Baudrate Switches = 0x03) then permission to override the baudrate and/or MacId is granted by bits 0 and 1 in this parameter.

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Set	Parameter Value	USINT	DeviceNet Override	0x00 = No Override 0x01 = DeviceNet 0x02 = Serial Link 0x03 = Both
2	Get	Link Path Size	USINT		0x00 No Link Path
3	Get	Link Path	ARRAY		<NULL> No Link Path
4	Get	Descriptor	WORD		0x0000
5	Get	Data Type	EPATH		0xC6
6	Get	Data Size	USINT		0x01

2.8.7 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Modifies an attribute value

2.8.8 Effect on Behavior

The IO configuration parameter has three(3) possible values in the range [0..2]. They correspond to Serial IO, Hardware IO, and Reserved . Changing the IO configuration is reflected in the values of other attributes. The Reserved space corresponds to an unimplemented alternate configuration which is undefined.

Attribute Name	Attribute Address	IO Configuration		
		0 Serial Link	1 Hardware	Reserved
Poll produced connection size	{5,2,7}	0..8	1	
Poll consumed connection size	{5,2,8}	0..8	1	
COS/Cyclic produced connection size	{5,4,7}	0..8	1	
Poll produced path	{5,2,14}	…24.64…	…24.65…	
Poll consumed path	{5,2,16}	…24.78…	…24.79…	
COS/Cyclic produced path	{5,4,14}	…24.64…	…24.65…	
Identity Object Status.2 “Out of Box” Configuration Status	{1,1,5}	Depends on Serial Link Diagnostic Action {6A,1,7}	Depends on Output Fault/Idle Behavior {9,X,5-8}	

2.9 Acknowledge Handler Object

The Acknowledge Handler Object is used to manage the reception of message acknowledgements.

2.9.1 Class Attributes

No class attributes are supported.

2.9.2 Class Services

No class services are supported.

2.9.3 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Set	Acknowledge Timer	UINT	Time to wait for acknowledge before resending	[1..65535] ms. 0 is invalid default = 16
2	Get/Set	Retry Limit	USINT	Number of Ack Timeouts before retry limit reached event	[0..255] default = 1
3	Get	COS Producing Connection Instance	UINT	Connection Instance which contains the path of the producing IO application	0x0004 Predefined COS Connection

2.9.4 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Modifies an attribute value

3 Vendor Specific Objects

3.1 Power Sense Object (0x64)

The Power Sense Object is used in conjunction with the Discrete Inputs for power sensing, PS2 and PS1. There is a single instance containing a single Boolean attribute.

3.1.1 Class Attributes

None

3.1.2 Class Services

None

3.1.3 Instance Attributes

Attribute ID	Access Rule	Name	Data Type	Description	Value
1	Get/Set	Value	BOOL	Ps_Polarity	0 = N Polarity 1 = P Polarity

3.1.4 Instance Services

Service Code	Service Name	Description of Service
0E	Get_Attribute_Single	Returns the contents of the specified attribute
10	Set_Attribute_Single	Modifies an attribute value

3.2 Serial Link Object (0x6A)

Publication #2200092 DN-SLIP DeviceNet Serial Interface Protocol Functional Specification describes the Serial Link Object. That text is replicated here for convenience.

This object provides the support for a serial link connection, attached to an asynchronous UART device. There should be one instance of this object for each physical or software simulated UART.

3.2.1 Class Attributes

Attribute ID	Need in Implementation	Access Rule	Name	DeviceNet Data Type	Description of Attribute
1	Conditional	Get	Revision	UINT	Revision of this object Current Value = 0x0001
2	Conditional	Get	Max Instance	UINT	Maximum instance number
3-7	Optional	These attributes are optional and described in Volume II chapter 5 of the DeviceNet Specification			

3.2.2 Instance Attributes

Attribute ID	Need in Implementation	Access Rule	Name	DeviceNet Data Type	Description of Attribute
1	Required	Get	Baudrate	UINT	Nominal Baudrate in bits per second. Default is: 9534 = 0x253E
2	Required	Get	Mode	USINT	'A' – Asynchronous (Default) 'S' – Synchronous
3	Required	Get	Parity	USINT	'N' – No Parity (Default) 'E' – Even Parity 'O' – Odd Parity '0' – Zero Stick Parity '1' – One Stick Parity
4	Required	Get	Data Bits	USINT	'5' – 5-bit '6' – 6-bit '7' – 7-bit '8' – 8-bit (Default)
5	Required	Get	Stop Bits	USINT	'1' – One Stop Bit (Default) '2' – Two Stop Bits 'H' – One and ½ Stop Bits
6	Required	Get	Check Sum	USINT	'0' – 2's Complement (Default) '1' – Straight Sum '-' – No Checksum
7	Required	Get/Set	Diagnostic Action	USINT	'0' – No Diagnostic Action '1' – Echo '2' – Transmit Character '3' – Transmit Frame
8	Required	Get/Set	Diagnostic Character	USINT	Any value in the range [0..255]
9	Required	Get/Set	Link Timer	UINT	Any value in the range [0..65535] in milliseconds
10	Optional	Get/Set	Rx Framing Error	USINT	[0..255]
11	Optional	Get/Set	Rx Data Overrun	USINT	[0..255]
12	Optional	Get/Set	Serial Transactions	UINT	[0..65535]
13	Optional	Get	Serial Status Register	USINT	SSR.7 – Invalid Level SSR.1 – Data
14	Optional	Get	Rx Buffer	ARRAY	20 bytes
15	Optional	Get	Rx Buffer Length	USINT	[1..20]

Each of the instance attributes in the range[1..9] is required in an implementation. A product is not required to support all possible values for an attribute. If only one alternative is supported then the access rule may be restricted to Get. If none of the attributes are settable, then they just document the implementation. It is also possible that different rules may apply for Explicit Requests from the Serial Link than Explicit Requests from the DeviceNet Link.

3.2.3 Common Services

The Serial Link Object provides the following Common Services

Service Code	Need in Implementation		Service Name	Description of Service
	Class	Instance		
0x0E	Conditional	Required	Get_Attribute_Single	Returns the content of the specified Attribute
0x10	No	Conditional	Set_Attributte_Single	Modifies the value of an Attribute

3.2.4 Diagnostic Action Behavior

Each of the following diagnostic action modes is entered by setting the value of the Diagnostic Action attribute to a particular non-zero ASCII value with an Explicit Message.

3.2.4.1 Diagnostic Action 1: Echo

In this state all characters consumed by the Serial Link Receiver are echoed to the Serial Link Transmitter as expeditiously as possible. The Serial Link Object will stay in this state until the Diagnostic Action attribute changed.

3.2.4.2 Diagnostic Action 2: Transmit Diagnostic Character

In this state, the Serial Link Transmitter will send the Diagnostic Character continuously at a rate determined by the Serial Link Timer, which is expressed in units of milliseconds. An End Delimiter (0xC0) will stop the continuous transmission. While stopped, any other character will cause the continuous transmission to restart.

3.2.5 Serial Port Status/Module Status LED Behavior

The DN-410-0.5 half-size board can detect an invalid level on the RS-232 Transceiver. It will signal this condition as a Major Recoverable Fault. The Module Status LED will go to a solid red condition. When the transceiver detects a valid RS-232 level the Module Status LED will return to its normal solid green condition. The occurrence of the transceiver fault is latched in bit seven (7) of the Serial Status Register {0x6A,1,13}. This bit is cleared when the attribute is read with an explicit message. As manufactured the full size DN-410 does not have this capability. The DN-400 and DN-TC400 devices ignore the invalid level condition on the serial port.

3.3 UART Object

This object is used for diagnostic purposes during development and may not be supported in released products. This object allows control and manipulation of the serial port from DeviceNet and must be used with extraordinary care.

3.4 Non-Volitile Test Object

This object is used for diagnostic purposes, and factory configuration. This object allows control and manipulation of the non-volitile memory from DeviceNet and must be used with extraordinary care.

4 Serial Link Predefined Connections

The Serial Link implements a point-to-point link through an asynchronous UART and an RS-232 transceiver. The DN-400 family devices implement the server endpoint of a predefined Explicit Connection and the server endpoint of predefined IO Connection. These connections are not represented in the object model, are present at all times, and do not need to be allocated or released. The connection endpoint outside the DN-400 family device is assumed to be the master and can have any MacId. The DN-400 family device will accept message identifiers from any MacId and will construct response identifiers with its own MacId. Communication on the Serial Link is further described in document #2200092 DN-SLIP Functional Specification.

4.1 Explicit Message Connection and the Serial Link Message Router

The Serial Link has an Explicit Message Router, which is separate from the one used to access the Object Model over DeviceNet. This Serial Link Message Router can access the attributes of the DeviceNet Object, the attributes of the Parameter Object, and the Factory Lock and Factory Unlock services of the Factory Test Object.

Object	Class Id	Notes
DeviceNet Object	0x03	Supports GET on Attributes [1..9] Supports SET on Attributes [1..2] with permission
Parameter Object	0x0F	Same access rules as over DeviceNet
Factory Test Object		Supports the Factory Unlock Service Supports the Factory Lock Service

4.2 IO Message Connection

Over the Serial Link there is a predefined POLL IO connection. The consumed and produced sizes are configurable in the range of zero(0) to eight(8) bytes. The connection consumes a DeviceNet POLL Response message identifier (Group 1, Message Id 15, any source MacId), places the data in a memory array, which is used to CONSTRUCT a DeviceNet POLL Response, and produces a DeviceNet POLL Request message (Group 2, My MacId, Message ID 5) with the data from a memory array containing the last data from a DeviceNet POLL Request. In some cases, when there is no valid data, a null message can be produced. Null messages should be consumed and interpreted as “No valid data is available”.

5 Configuration

5.1 Firmware

The firmware for the DN-400 family is stored in FLASH program memory inside the single chip processor (Atmel T89C51CC01). This memory may be electrically erased and reprogrammed multiple times. It is even possible to upgrade a device in the field. This is done with the six-pin serial cable. If pins 1 and 6 are shorted together the processor will enter the boot loader mode. A free software package from Atmel called FLIP can be used to download new firmware. On some newer versions of the Atmel processor it is necessary to make sure that the "Boot Loader Jump Bit" is cleared. Failure to do so will result in the repeated execution of the boot loader instead of the application.

5.2 Non-Volitile Data Configuration

All of the configuration parameters have default values defined in the firmware that are used when the processor is programmed for the first time. As various changes are made to these values in RAM they are updated in the Non-Volitile Data Memory. At some point in time a Factory Lock service is performed by the Factory Test Object. It saves all of the configuration parameters twice. Once to reflect the values that are currently being used and the second time to record the "As Shipped" value of each parameter so that a RESET service to the Identity Object can return the device as close as possible to the "Out of Box" configuration.