



User Manual for the
HE693DNT750

DeviceNet™ Slave

Third Edition
22 June 2000

MAN0213-03

PREFACE

This manual explains how to use the Horner APG's HE693DNT750 DeviceNet Slave.

Copyright (C) 2000 Horner APG, LLC., 640 North Sherman Drive, Indianapolis, Indiana 46201. All rights reserved. No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior agreement and written permission of Horner APG, LLC.

All software described in this document or media is also copyrighted material subject to the terms and conditions of the Horner Software License Agreement.

Information in this document is subject to change without notice and does not represent a commitment on the part of Horner APG.

DeviceNet is a trademark of Open DeviceNet Vendors Association (ODVA).
LogicMaster, LM90, and SNP are trademarks of GE Fanuc

For user manual updates, contact Horner APG, Technical Support Division, at (317) 916-4274 or visit our website at www.heapg.com.

LIMITED WARRANTY AND LIMITATION OF LIABILITY

Horner APG, LLC. ("HE-APG") warrants to the original purchaser that the DeviceNet™ Slave manufactured by HE is free from defects in material and workmanship under normal use and service. The obligation of HE under this warranty shall be limited to the repair or exchange of any part or parts which may prove defective under normal use and service within two (2) years from the date of manufacture or eighteen (18) months from the date of installation by the original purchaser whichever occurs first, such defect to be disclosed to the satisfaction of HE after examination by HE of the allegedly defective part or parts. THIS WARRANTY IS EXPRESSLY IN LIEU OF ALL OTHER WARRANTIES EXPRESSED OR IMPLIED INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE AND OF ALL OTHER OBLIGATIONS OR LIABILITIES AND HE NEITHER ASSUMES, NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR HE, ANY OTHER LIABILITY IN CONNECTION WITH THE SALE OF THIS DeviceNet™ Slave. THIS WARRANTY SHALL NOT APPLY TO THIS DeviceNet™ Slave OR ANY PART THEREOF WHICH HAS BEEN SUBJECT TO ACCIDENT, NEGLIGENCE, ALTERATION, ABUSE, OR MISUSE. HE MAKES NO WARRANTY WHATSOEVER IN RESPECT TO ACCESSORIES OR PARTS NOT SUPPLIED BY HE. THE TERM "ORIGINAL PURCHASER", AS USED IN THIS WARRANTY, SHALL BE DEEMED TO MEAN THAT PERSON FOR WHOM THE DeviceNet™ Slave IS ORIGINALLY INSTALLED. THIS WARRANTY SHALL APPLY ONLY WITHIN THE BOUNDARIES OF THE CONTINENTAL UNITED STATES.

In no event, whether as a result of breach of contract, warranty, tort (including negligence) or otherwise, shall HE or its suppliers be liable of any special, consequential, incidental or penal damages including, but not limited to, loss of profit or revenues, loss of use of the products or any associated equipment, damage to associated equipment, cost of capital, cost of substitute products, facilities, services or replacement power, down time costs, or claims of original purchaser's customers for such damages.

To obtain warranty service, return the product to your distributor with a description of the problem, proof of purchase, post paid, insured and in a suitable package.

ABOUT PROGRAMMING EXAMPLES

Any example programs and program segments in this manual or provided on accompanying diskettes are included solely for illustrative purposes. Due to the many variables and requirements associated with any particular installation, Horner APG cannot assume responsibility or liability for actual use based on the examples and diagrams. It is the sole responsibility of the system designer utilizing the to appropriately design the end system, to appropriately integrate the DeviceNet™ Slave and to make safety provisions for the end equipment as is usual and customary in industrial applications as defined in any codes or standards which apply.

Note: The programming examples shown in this manual are for illustrative purposes only. Proper machine operation is the sole responsibility of the system integrator.

Revisions to this Manual

This version (MAN0213-03) of the DeviceNet™ User Manual contains the following revisions, additions, and deletions:

1. Revised Section 1.1.
2. Revised Section 3.2.
3. Revised Section 3.3.1
4. Revised Section 3.3.2. Deleted Table 3.3.
5. Deleted Section 3.3.4: Invocation. Replaced with new Section 3.3.4: Configuration Procedures.
6. Deleted Section 3.3.5 and Section 3.3.6.
7. Deleted all sections in Section 4 and replaced with new. Replaced with new Sections 4.1 and 4.2.
8. Deleted Sections 5.1 and 5.2. Replaced with Section 5.1: Explicit Connections and Section 5.2: Connection Based Explicit Messages.
9. Deleted Section 5.3. Replaced with Section 5.3: Success Response Messages.
10. Deleted Section 5.4. Replaced with Section 5.4: Error Response Messages.
11. Renamed Table 5.1 to Table 5.6. Re-numbered Table 5.2 as Table 5.5.

TABLE OF CONTENTS

PREFACE.....3

LIMITED WARRANTY AND LIMITATION OF LIABILITY4

ABOUT PROGRAMMING EXAMPLES4

TABLE OF CONTENTS7

CHAPTER 1: INTRODUCTION9

 1.1 Overview..... 9

 1.2 HE693DNT750 Features10

 1.3 Technical Specifications.....10

CHAPTER 2: INSTALLATION11

 2.1 Connectors11

 2.1.1 RS-485 Connector11

 2.2.2 DeviceNet I/O Connector.....11

 2.2 LED Indicators12

CHAPTER 3: CONFIGURATION13

 3.1 General.....13

 3.2 Module Configuration13

 3.2.1 HE693DNT750 DeviceNet Parameters.....13

 3.2.2 HE693DNT750 Implementation.....13

 3.3 The DN750CFG.EXE Utility.....14

 3.3.1 Overview.....14

 3.3.2 Requirements.....14

 3.3.3 Installation.....14

 3.3.4 Configuration Procedures.....15

 3.4 Configuring the PLC.....18

 3.5 Configuring the PLC using the Hand Held Programmer (HHP)18

 3.6 Configuring the PLC using the LogicMaster (LM90).....19

CHAPTER 4: DEVICENET TO PLC I/O MAPPING (POLLED CONNECTION)21

 4.1 Polled Connection Overview.....21

 4.2 Programmable Parameters21

CHAPTER 5: DEVICENET TO PLC I/O MAPPING (EXPLICIT CONNECTION).....23

 5.1 Explicit Connection.....23

CHAPTER 1: INTRODUCTION

1.1 Overview

The HE693DNT750 (DNT750) is a DeviceNet Slave Module for use with a GE Fanuc Series 9030 Programmable Logic Controller (PLC). The DNT750 provides DeviceNet slave functionality to the PLC when connected to a DeviceNet network.

The DNT750 is a Group 2 Only Server. It is a slave device, which communicates using the DeviceNet Predefined Master/Slave Connection Set. It does not support UCM.

Within the Predefined Master/Slave Connection Set, the DNT750 supports "Explicit" and "Polled" connections. Bit-Strobed, Cyclic and Change of State connections are not supported.

Included with the HE693DNT750 module is a diskette containing the HE693DNT750 DeviceNet™ Slave/Series 90-30 Configuration Utility, `DN750CFG.EXE`. *It is important to note that the HE693DNT750 must be configured (using `DN750CFG.EXE`) before the PLC is configured (using `LogicMaster™` or `Versa Pro™`).*

Note: Version 2.0 of the `DN750CFG.EXE` Configuration Utility is Windows-based. Earlier versions of `DN750CFG.EXE` prior to Version 2.0 are DOS-based.

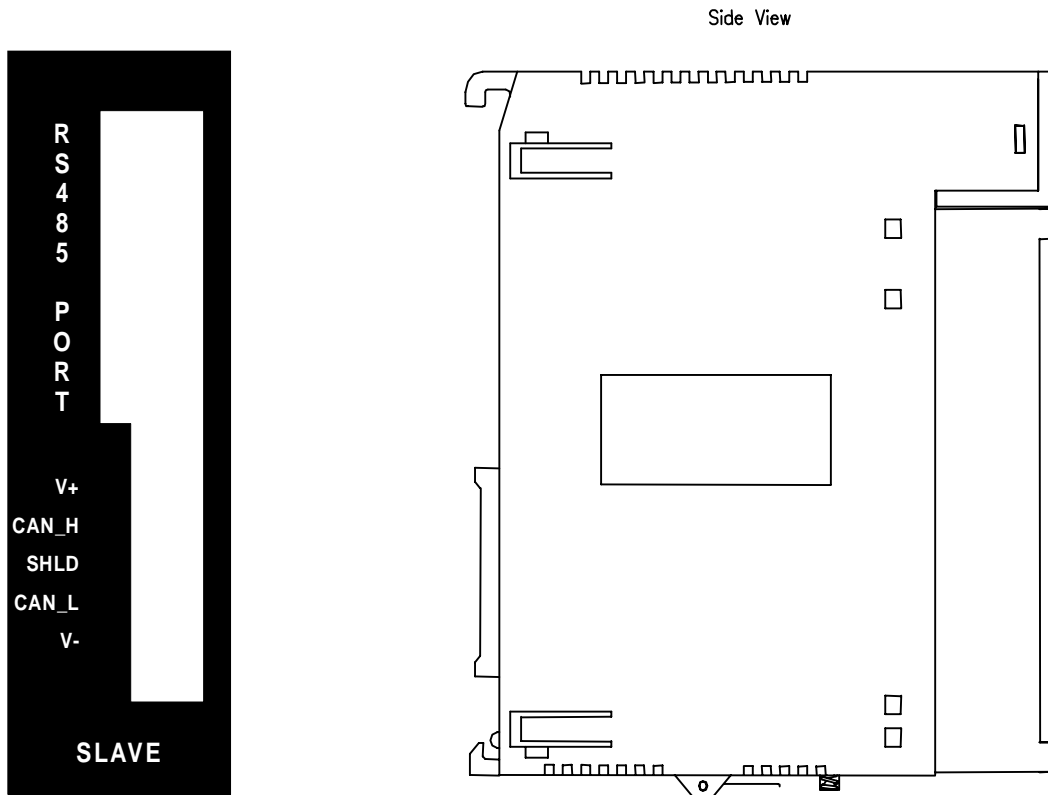


Figure 1.1 – HE693DNT750 Module

1.2 HE693DNT750 Features

The HE693DNT750 supports 125K, 250K, and 500K DeviceNet baud rates. Also supported are both Polled and Explicit Messaging with full fragmentation.

Using Polled Messaging, the attached PLC's %I, %Q, %AI, and %AQ registers are directly accessible. Using Explicit Connection, the DeviceNet Master can access the PLC's %I, %Q, %R, %AI, %AQ, %T, and %M registers.

During operation, the status of the module and the bus are indicated by two front panel LEDs.

Module operational parameters (MACID, baud rate, size of Polled Message) can be configured using the separate configuration program, DN750CFG.EXE. This program also allows individual module configurations to be saved to or loaded from disk, or printed to a printer for hard copy.

IMPORTANT: The HE693DNT750 must be configured (using DN750CFG.EXE) before the PLC is configured (using LogicMaster™ and Versa Pro™).

The HE693DNT750 does not accept SNP or LogicMaster codes. The HE693DNT750 must be configured using the DN750CFG.EXE program.

IMPORTANT: Do **NOT** attempt to "talk to" the HE693DNT750 using LogicMaster or SNP Protocol. No physical damage will result, but information in the HE693DNT750 could be overwritten and could possibly require a factory update to be returned to normal operating conditions

1.3 Technical Specifications

Table 1.1 - Technical Specifications			
PLC Power Load Specifications			
Parameter	Minimum	Maximum	Units
+5Vdc (LOGIC)	0	175	mA
+24Vdc (RELAY)	0	0	mA
+24Vdc (ISOLATED)	0	0	mA
Devicenet™ Network Specifications			
Parameter	Minimum	Maximum	Units
DeviceNet Power Voltage	11	25	V
DeviceNet Power Load		15	mA
DeviceNet Signal Baud Rate	125	500	KHz
DeviceNet Signal Driver Fanout	0	63	Devices
Environmental Specifications			
Parameter	Minimum	Maximum	Units
Operating Temperature	0	60	Degrees Celsius
Storage Temperature	-40	85	Degrees Celsius
Humidity (Non-condensing)	10	95	% Relative Humidity
Devicenet™ Baud Rate			
DeviceNet Baud Rate	Maximum Distance		
125KHz	1500 Feet		
250KHz	750 Feet		
500KHz	300 Feet		

CHAPTER 2: INSTALLATION

2.1 Connectors

2.1.1 RS-485 Connector

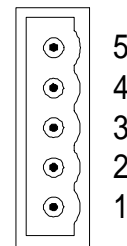
The RS-485 configuration port consists of a 15-pin female D connector with the following pin descriptions. It is compatible with the standard programming cable/adaptor, the HE693SNPCBL.

Table 2.1 – RS-485 Connector		
Pin	Signal	Description
1	FGND	Frame Ground
2	DCD-	Data Carrier Detect -
3	DCD+	Data Carrier Detect +
4	-HHP	Hand Held Programmer Detect
5	+5V	RS485-RS232 Adapter Power
6	RTS-	Request to Send -
7	GND	Signal Ground
8	CTS+	Clear to Send +
9	RT	Receiver Termination
10	RXD-	Received Data -
11	RXD+	Received Data +
12	TXD-	Transmit Data -
13	TXD+	Transmit Data +
14	RTS+	Request to Send +
15	CTS-	Clear to Send -

2.2.2 DeviceNet I/O Connector

The DeviceNet I/O connector consists of a 5-pin removable screw terminal with the following terminal descriptions:

Table 2.2 – I/O Connector		
Pin	Signal	Description
1	V-	Power -
2	CAN_L	Signal -
3	Drain	Shield
4	CAN_H	Signal +
5	V+	Power +



A user supplied 120 Ω resistor is needed for termination at **each end** of the network cabling. Refer to the DeviceNet cabling specifications for proper location of the terminating resistor.

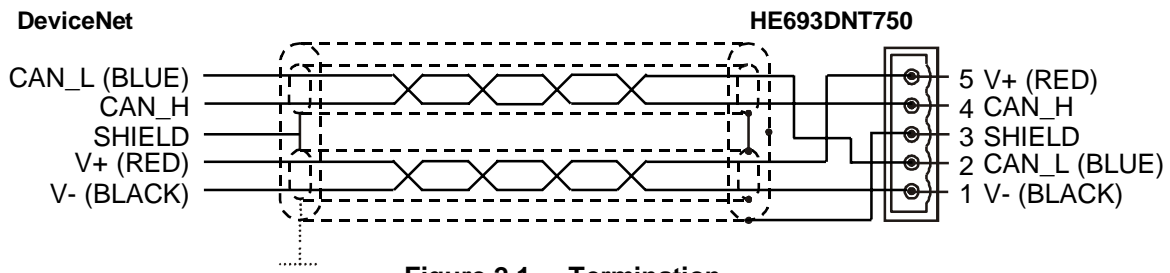


Figure 2.1 - Termination

2.2 LED Indicators

The HE693DNT750 provides two bi-color (Red/Green) diagnostic LEDs on its front panel. Tables 2.3 and 2.4 describe the LED indications that depict the status of the module and the network.

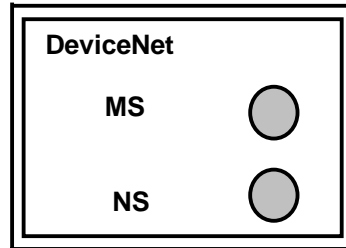


Figure 2.2 – LED Front Panel Indicators

Module Status LED is	State	Condition
Off	No Power	There is no power applied to the module.
Green	Device Operational	The module is operating normally.
Flashing Green	Device in Standby	The module has no configuration.
Flashing Red	Minor Fault	Recoverable Fault
Red	Unrecoverable Fault	The module has an unrecoverable fault.
Flashing Red-Green	Device Self Testing	The module is in Self Test.

Network Status LED is	State	Condition
Off	Not powered or not On-line	The module may not be powered. Module is not on-line. Module has not completed Dup_MAC_ID test.
Green	Module On-line and connected. Network OK	The module is on-line and has connections in the established state.
Flashing Green	Module On-line but not connected	The module has passed the Dup_MAC_ID test and is on-line, but has no established connections to other nodes.
Flashing Red	Connection Time-Out	One or more Connections have Timed-Out.
Red	Critical Link Failure	Failed communication device.
Flashing Red-Green	Device Self Test	Device is in self test
Flashing Red	Device in Standby	The module has no configuration

CHAPTER 3: CONFIGURATION

3.1 General

Before the HE693DNT750 can be used as a I/O module, it must first be properly configured. This consists of two steps: Module Configuration and PLC Configuration.

IMPORTANT: The module configuration must occur **prior** to configuring the PLC.

3.2 Module Configuration

Module configuration is accomplished using the `DN750CFG.EXE`, Windows-based configuration utility. The utility program provides ability to download/upload configuration to/from DNT750 module. The configuration can also be saved to a configuration file and reloaded when needed. The utility program is described in this section. This section also enlists the information a DeviceNet Scanner will need regarding DNT750 slave device.

Note: Version 2.0 of the `DN750CFG.EXE` Configuration Utility is Windows-based. Earlier versions of `DN750CFG.EXE` - prior to Version 2.0 - are DOS-based.

3.2.1 HE693DNT750 DeviceNet Parameters

Many scanner devices will require the following information to be entered into their Scan List:

HE693DNT750 Vendor ID:	86
HE693DNT750 Product Code:	6
HE693DNT750 Device Type:	0

3.2.2 HE693DNT750 Implementation

This implementation provides a subset of the complete DeviceNet™ specification.

Supported points:

"Group 2 Only" slaves via the "Predefined Master/Slave Connection Set".
"Polled I/O Messaging" with Fragmentation
"Explicit Messaging" with Fragmentation

Non-supported points:

"Explicit Unconnected Message Manager" (UCMM).
"Bit-Strobed I/O Messaging".
"Cyclic Connection Messaging"
"Change of State Messaging"

3.3 The DN750CFG.EXE Utility

3.3.1 Overview

IMPORTANT: The HE693DNT750 must be configured before the PLC is configured.

A Windows-based, stand-alone program, DN750CFG.EXE, is provided for configuration of the HE693DNT750 module. This program allows the User to configure the amount of and type of data transferred by the HE693DNT750 using the Polled Connection, as well as the HE693DNT750's MACID (Network Address) and Baud Rate. The user can program off line, without a direct connection to a HE693DNT750 module. Finished configurations may be saved to disk, then re-called and downloaded to a module later. This allows the user to develop a configuration locally, and then program modules at a remote site using a laptop PC.

3.3.2 Requirements

The configuration utility DN750CFG.EXE, Beginning with Version 2.0, is a 32-bit Windows application. It can be installed and run on most Windows 95, Windows 98 and Windows NT based PC. DN750CFG.EXE will not run under DOS or Windows 3.1.

In order for the configuration utility to communicate with the DNT750 module – to upload, download or verify configuration – a COM port on the PC should be connected to the RS485 port on the DNT750 module. For this purpose, the user also needs a serial cable and a SNP adapter.

Description	Horner Electric Part Number
RS232-to-SNP adapter	HE693SNP232
9-pin female-to-female straight through cable	HE693CBLSNP
Cable Set (contains both cable and adapter)	HE693SNPCBL

3.3.3 Installation

The diskette distributed with the HE693DNT750 module contains the HE693DNT750 DNT Interface Configuration Utility (DN750CFG.EXE) setup program Setup.Exe.

1. Insert the diskette into A: (or B:) drive.
2. As with any installation program, it is a good practice to make a back-up copy of the installation diskette.
3. Execute the setup program provided on the diskette (double-click on the Setup.Exe program icon). The setup program guides through rest of the installation procedure.
4. The Dn750Cfg utility program is installed in the directory chosen during setup.

DN750CFG can work off line, with no module attached. However, in order to configure a HE693DNT750 module, the user must first connect it to the PC:

1. With the rack power off, plug the HE693DNT750 module into an open I/O slot of a GE Fanuc Series 90-30 rack.
2. Plug the SNP adapter (P/N HE693SNP232) into the 15-pin female RS485 port on the HE693DNT750 module.

3. Plug one end of the 9-pin female D to 9-pin female “straight-through” cable (P/N HE693CBLSNP) to the 9-pin end of the SNP Adapter. Connect the free end of the cable to the desired COM port on the PC. If the PC is equipped with 25-pin COM ports, use a standard 25-pin female to 9-pin male adapter.
4. Power up the PLC rack.

3.3.4 Configuration Procedures

Double-click the DN750CFG Configuration Utility icon in the installation folder. The following screen appears.

Note: The title of the screen indicates that the configuration file is untitled. It remains as such until the configuration is saved with a unique name. Each DNT750 slave device on the network needs to be individually configured. After creating or modifying a configuration for a particular slave device, it is recommended to save the file with a unique name for future reference.

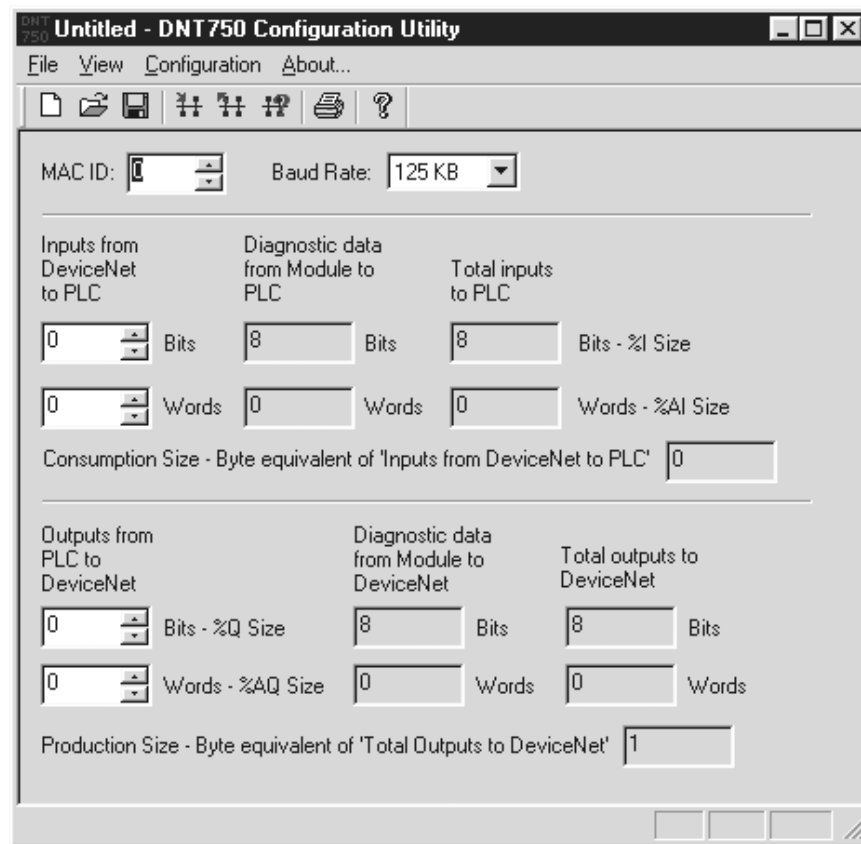


Figure 3.1 – Untitled Default Configuration Screen

1. Type a unique DeviceNet Address into the MAC ID block and press enter.
2. Using pull-down menu, select one of three possible baudrates that can be used in DeviceNet networks. 125K, 250K or 500K.

3. Configure the following parameters: Refer to Figures 3.2 and 3.3.
- Enter the **Inputs from DeviceNet to PLC** in bits and words. Bits need to be in multiple of eights. If they are not entered as such, a dialog appears indicating that the number will be adjusted to the lower multiple.
 - The **Diagnostic data from Module to PLC** is shown in bits and words. This is constant and cannot be edited by the user.
 - The **Total inputs to PLC** is shown in bits and words. This value is the total of “Inputs from DeviceNet to PLC” and “Diagnostic data from Module to PLC”. This value cannot be directly edited by the user. Note that the Total inputs to PLC in bits is the %I size; and the Total Inputs to PLC in words is the %AI size. These values (%I size and %AI size) are needed when configuring the PLC as described in the next section. PLC backplane limits the total byte equivalent of %I size and %AI size to less than or equal to 254 bytes.
 - Consumption Size** is the total byte equivalent of Inputs from DeviceNet to PLC. This value is for information only, and cannot be edited by the user directly.

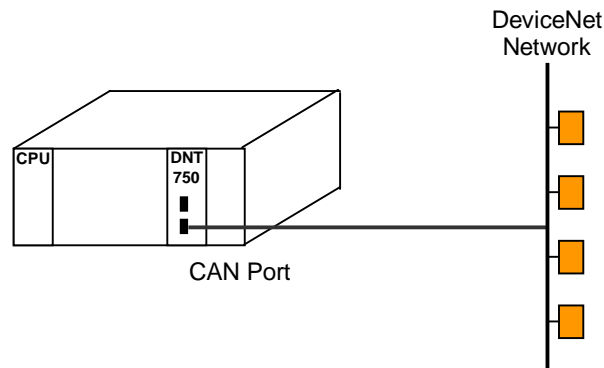


Figure 3.2 – Overview of Setup

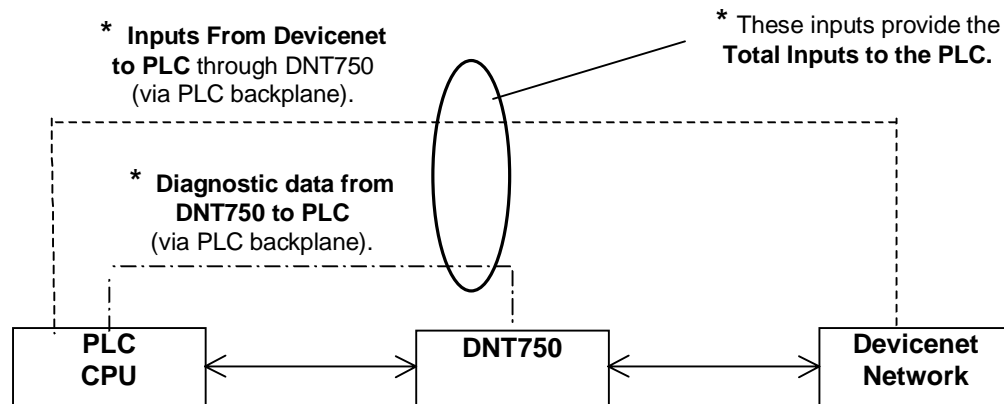


Figure 3.3 – Close-Up View of Inputs

4. Configure the following parameters: Refer to Figures 3.2 and 3.4.
 - a. Enter the **Outputs from PLC to DeviceNet** in bits and words. Bits need to be in multiple of eights. If they are not entered as such, a dialog appears indicating that the number will be adjusted to the lower multiple. Note that the Outputs from PLC to DeviceNet in bits is the %Q size; and the Outputs from PLC to DeviceNet in words is the %AQ size. These values (%Q size and %AQ size) are needed when configuring the PLC as described in the next section. PLC backplane limits the total byte equivalent of %Q size and %AQ size to less than or equal to 254 bytes.
 - b. The **Diagnostic data from Module to DeviceNet** is shown in bits and words. This is constant and cannot be edited by the user.
 - c. The **Total Outputs to DeviceNet** is shown in bits and words. This value is the total of Outputs from PLC to DeviceNet and Diagnostic data from Module to DeviceNet and cannot be edited directly by the user.
 - d. **Production Size** is the byte equivalent of Total Outputs to DeviceNet. This value is for information only, and cannot be edited by the user directly.

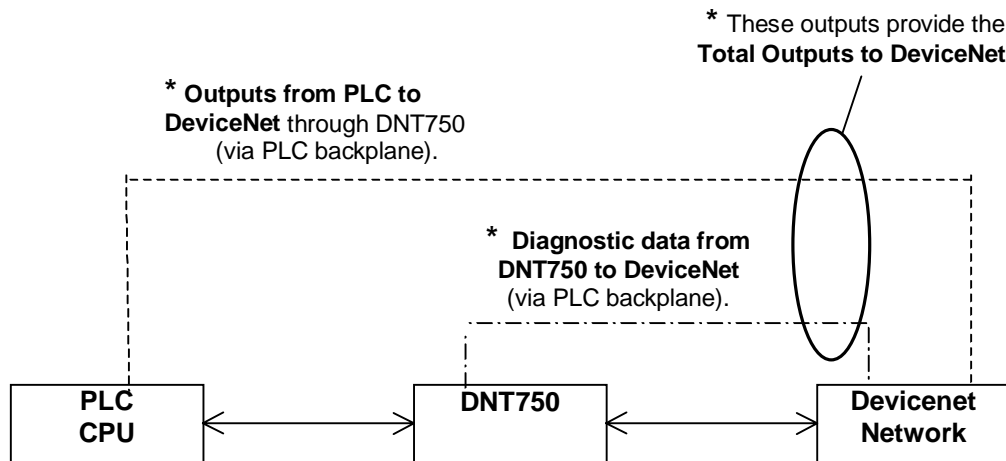







Figure 3.4 – Close-Up View of Outputs

5. You may want to do some documentation regarding the configuration using the Configuration | Project menu option.
6. After entering the above parameters, it is recommended to save the configuration using File | Save menu option (or Save  button on the toolbar). If this is a new configuration, a Save As dialog appears, so that you may specify the file name using which this configuration will be saved.

7. In order to communicate (download/upload/verify configuration) with the DNT750 module, connect the DNT750 serial port to PC's available COM port as described in installation instructions in the previous subsection. Then select the COM port chosen using Configuration | Select Com Port menu option, so that the utility program knows which Com port to use.

Note: While trying to establish communications during **upload, download or verification**, an error message may appear. If this happens, try again. If the error persists, (a) Check the serial cable link between the module and PC (b) Make sure that no other application on the PC is already using the selected Com port (c) power-cycle the PLC.

8. Using Configuration | Download option (or  Download button on the toolbar), the configuration can now be downloaded to the DNT750 module.
9. If you need, you may verify if the configuration in the file is same as Module configuration using Configuration | Verify menu option (or  Verify button on the toolbar).
10. If a copy of the current configuration in DNT750 module is needed, the user can **upload** the configuration from the DNT750 module to the PC using Configuration | Upload menu option (or  Upload button on the toolbar).
11. You may take a hard copy of the configuration using the File | Print menu option (or  Print button on the toolbar). Using the File | Print Options menu option, you may select whether you want to print Project Summary and/or Module Configuration.

3.4 Configuring the PLC

IMPORTANT: The HE693DNT750 must be configured (using DN750CFG.EXE) **before** the PLC is configured.

The PLC is configured using either a Hand Held Programmer or LogicMaster (LM90) or Versa Pro™.

3.5 Configuring the PLC using the Hand Held Programmer (HHP)

The following sequence may be used to configure the PLC rack for an HE693DNT250 module via the Hand-Held Programmer (HHP):

1. With the rack power off, plug the HE693DNT250 module into an open I/O slot of a PLC. Connect the HHP.
2. Power up the PLC rack.
3. On the HHP, press (<MODE> , <4> , < ENTER>) to get into configuration mode.
4. Press <DOWN ARROW> repeatedly until the HHP's display shows the slot containing the HE693DNT250 module.
5. If display does not show the slot as <EMPTY> , press , <ENTER> to empty it.
6. Turn power off. Then, turn power on again.

7. Press <READ>, <ENTER> Note - the %I reference, %I size, %Q reference, %Q size, %AI reference, %AI size, %AQ reference and %AQ size parameters should be set to match those observed on the global screen when the module configuration was performed.

Global Parameter	Value Range	Default	Description
%I Reference Address	0 - 9993	1	First address (offset) of the block of data
%I Size	*	-	Total number of configured registers
%Q Reference Address	0 - 9993	1	First address (offset) of the block of data
%Q Size	*	-	Total number of configured registers
%AI Reference Address	0 - 9999	1	First address (offset) of the block of data
%AI Size	*	-	Total number of configured registers
%AQ Reference Address	0 - 9999	1	First address (offset) of the block of data
%AQ Size	*	-	Total number of configured registers

* These values are not modifiable - they indicate the total size of the corresponding %I, %Q, %AI and %AQ register blocks that have been configured into the HE693DNT750 module.

The sum of all inputs, %I & %AI registers, can not exceed 254 bytes. The sum of all outputs, %Q & %AQ registers, can not exceed 254 bytes.

3.6 Configuring the PLC using the LogicMaster (LM90)

The following sequence may be used to configure the PLC rack for an HE693DNT250 module via LogicMaster

1. On the PC, invoke "LM90"
 2. At the first screen select Configuration Package (F2 key).
 3. Create or select a program folder in which to later store data, and press <enter>.
 4. At the Configuration Package menu select I/O configuration (F1 key).
 5. Use arrow keys to select the slot in which the HE693DNT750 is installed.
- Note: If any module already exists in the desired slot, delete the old module by using the <shift F4>.
6. Press F8 (Other) to select the module type.
 7. Select F3 (Foreign Module).
 8. Press F10 (Zoom) to view and edit the module's parameters.
 9. Using the information obtained from DN750CFG.EXE, set the %I Reference, %I Size, %Q Reference, %Q Size, %AI Reference, %AI Size, %AQ Reference and %AQ Size parameters.
 10. Set the Byte 1 parameter on right side of screen to "1". Bytes 2 through 16 should remain.
 11. Press the <escape> key TWICE.
 12. Press F9 to select the Download Menu.
 13. Press F2 to store the configuration to the PLC.
 14. Press <enter> to begin store and overwrite.
 15. Power down the PLC rack.
 16. Power up the PLC rack.

NOTE: When *reconfiguring* the module it is important that the old module be *deleted* (not just altered) or the new configuration may not be recognized.

NOTES

CHAPTER 4: DEVICENET TO PLC I/O MAPPING (POLLED CONNECTION)

4.1 Polled Connection Overview

Using the polled connection, the scanner device can read from a group of the PLC's %Q and a group of the PLC's %AQ registers and can write to a group of the PLC's %I and a group of the PLC's %AI registers continuously. This allows for rapid data movement between the PLC's registers and the DeviceNet master.

The DNT750 generates two bytes of diagnostic data. The first diagnostic byte is added to the DeviceNet data stream that is sent to the DeviceNet scanner. This diagnostic byte is added in front of the balance of the data in the data assembly produced by the DNT750. The second byte is added to the data that is sent to the PLC. The PLC sees this byte as 8 additional %I registers.

There is a limitation imposed by the PLC backplane on the maximum amount of data that can be sent over the PLC backplane using the PLC's backplane I/O transfer method. This limit is 254 bytes. This limit directly impacts the DNT750's DeviceNet consumption and production sizes. The consumption size limit is 253 bytes and the production size limit is 255 bytes. These sizes include the diagnostic bytes generated by the DNT750.

4.2 Programmable Parameters

The DNT750 requires only six programmable parameters:

1. the DNT750's MACID or network address
2. the DeviceNet baud rate
3. the number of PLC %I registers that are to be received from DeviceNet scanner
4. the number of PLC %Q registers that are to be transmitted to the DeviceNet scanner
5. the number of PLC %AI registers that are to be received from DeviceNet scanner
6. the number of PLC %AQ registers that are to be transmitted to the DeviceNet scanner

The following formulas can be used to compute the Data Production and Consumption sizes for the polled connection:

Production size = (number of %Q bits / 8) + (number of %AQ * 2) + 1

Consumption size = (number of %I bits / 8) + (number of %AI * 2)

The input data assembly is a function of the number of PLC %I and %AI registers configured to be received by the DNT750. The output data assembly is a function of the quantity of %Q and %AQ registers configured to be transmitted by the DNT750. Bit type registers, %I and %Q registers, must be assigned in groups of 8. %AI and %AQ registers are 16 bit integer values, they each require 2 bytes. Because the actual data assembly is configuration dependent, the following generalizations can be assumed:

Within the DeviceNet data assemblies, the bit type register data is always sent or received first followed by the word type register data:

The Consumption (input) data assembly for the DNT750 is:

1. the number of PLC %I registers configured / 8, followed by,
2. the number of PLC %AI registers configured * 2.

The Production (output) data assembly for the DNT750 is:

1. 1 byte of diagnostic data generated by the DNT750 followed by,
2. the number of PLC %Q registers configured / 8 followed by,
3. the number of PLC %AQ registers configured * 2.

Data format of the diagnostic byte sent by the DNT750 to the PLC:

Bit 0 – set if a “BUS FAULT” is observed (loss of bus power or BUS_OFF)
Bit 1 – set if the polled connection is in the ESTABLISHED state
Bit 2 – set if the DNT750 is receiving “RECEIVE_IDLE” message packets
Bit 3 \
Bit 4 \
Bit 5 } Not used
Bit 6 /
Bit 7 /

Data format of the diagnostic byte sent by the DNT750 to the scanner:

Bit 0 – set if the PLC is in the “RUN” mode
Bit 1 \
Bit 2 \
Bit 3 \
Bit 4 } Not used
Bit 5 /
Bit 6 /
Bit 7 /

CHAPTER 5: DEVICENET TO PLC I/O MAPPING (EXPLICIT CONNECTION)

5.1 Explicit Connection

The explicit connection allows the master device to make requests of the DNT750. There are typically two cases where explicit messaging are used.

First, several explicit messages occur between the scanner and a DNT750 in order to establish a connection. These explicit messages are initiated automatically by the scanner and do not require any user interaction.

Second, explicit messaging allow the code running in the scanner's host device to send explicit messages in order to access specific data within the DNT750's PLC. The host device is usually a PLC or a PC of some sort. Because the overhead required for explicit messages is considerably greater than that of polled messaging, explicit messages should be used only to access data on an infrequent basis such as configuration or tuning parameters.

5.2 Connection Based Explicit Messaging

The DNT750 makes use of the standard DeviceNet format for Connection Based Explicit Messages as defined in the DeviceNet Specification Version 2.0 Section 4-2.5. In this message format the host application must specify:

1. MACID of the node being addressed,
2. Service Code,
3. Class ID,
4. Instance ID and,
5. Data as specified by the Service Code in item 1 above.

The DNT750 provides two new vendor specific service codes to allow access to the registers within the PLC:

1. Get PLC Register(s) – Service Code 0x32 (50 decimal)
2. Set PLC Register(s) – Service Code 0x33 (51 decimal)

The data format of the new service codes are outlined in the following table:

Table 5.1 – Service Codes		
Byte Position	Get Plc Registers	Set Plc Registers
0	MACID	MACID
1	Service Code (50 decimal)	Service Code (51 decimal)
2	Class ID (100 decimal)	Class ID (100 decimal)
3	Instance ID (see below)	Instance ID (see below)
4	Register Base Address LSB	Register Base Address LSB
5	Register Base Address MSB	Register Base Address MSB
6	Number of Registers LSB	Number of Registers LSB
7	Number of Registers MSB	Number of Registers MSB
8 - n	N/A	Data to store in the PLC registers

The Instance ID specifies which type of PLC registers are to be accessed:

Instance ID	PLC Register Type	Register Size
8 decimal	%R	16 bits
10 decimal	%AI	16 bits
12 decimal	%AQ	16 bits
16 decimal	%I	1 bit
18 decimal	%Q	1 bit
20 decimal	%T	1 bit
22 decimal	%M	1 bit

If the PLC Register Type indicates a single bit register type, then the Register Base Address parameter must be on a byte boundary (in the sequence 1, 9, 17, 25 etc. $(8n + 1)$), also, the Number of Registers parameter must be a multiple of 8. Note that only multiples of 8 bits can be read or written.

Example 1:

Suppose we want to read PLC registers %R300 through %R319:

Function	Value
MACID	0 – 63 decimal
Service Code	50 decimal
Class ID	100 decimal
Instance ID	8 decimal
Register Base Address LSB	300 decimal
Register Base Address MSB	
Quantity of Registers LSB	20 decimal
Quantity of Registers MSB	

In this case, either a Success Response Message or an Error Response Message will be returned. If a Success Response Message is returned, 40 bytes of data (20 words) will be included.

Example 2:

Suppose we want to write PLC registers %AI10 through %AI12 with the data 5, 10 & 15:

FUNCTION	VALUE
MACID	0 – 63 decimal
Service Code	51 decimal
Class ID	100 decimal
Instance ID	10 decimal
Register Base Address LSB	10 decimal
Register Base Address MSB	
Quantity of Registers LSB	3 decimal
Quantity of Registers MSB	
Data for %AI10 LSB	5 decimal
Data for %AI10 MSB	
Data for %AI11 LSB	10 decimal
Data for %AI11 MSB	
Data for %AI12 LSB	15 decimal
Data for %AI12 MSB	

In this case, either a Success Response Message or an Error Response Message will be returned. There is no returned data.

Example 3:

Suppose we want to read PLC registers %I17 through %I24:

FUNCTION	VALUE
MACID	0 – 63 decimal
Service Code	50 decimal
Class ID	100 decimal
Instance ID	16 decimal
Register Base Address LSB	17 decimal
Register Base Address MSB	
Quantity of Registers LSB	8 decimal
Quantity of Registers MSB	

In this case, either a Success Response Message or an Error Response Message will be returned. If a Success Response Message is returned, 1 bytes of data will be included, each bit of this data byte represent one of the specified 8 registers. The data for the register with the smallest address is in the LS bit position.

Example 4:

Suppose we want to write PLC registers %Q49 through %Q64 with the data:

Register	Data
%Q49	1
%Q50	0
%Q51	0
%Q52	1
%Q53	0
%Q54	1
%Q55	1
%Q56	0
%Q57	1
%Q58	0
%Q59	1
%Q60	0
%Q61	0
%Q62	1
%Q63	0
%Q64	1

FUNCTION	VALUE
MACID	0 – 63 decimal
Service Code	50 decimal
Class ID	100 decimal
Instance ID	18 decimal
Register Base Address LSB	49 decimal
Register Base Address MSB	
Quantity of Registers LSB	16 decimal
Quantity of Registers MSB	
Data for %Q49 – %Q56	0110 1001 binary
Data for %Q57 – %Q64	1010 0101 binary

In this case, either a Success Response Message or an Error Response Message will be returned. There is no returned data.

5.3 Success Response Messages

A Success Response Message is indicated by a response message with the following format:

BYTE POSITION	GET PLC REGISTERS	SET PLC REGISTERS
0	MACID	MACID
1	B2 Hex – 178 Decimal	B3 Hex - 179 Decimal
2 - n	Response Data	N/A

5.4 Error Response Messages

A number of different types of error messages may be returned by the DNT750. An error response is indicated by a response message with the following format:

BYTE POSITION	
0	MACID
1	94 Hex – 148 Decimal
2	Error Code Byte 1
3	Error Code Byte 2

BYTE 1	BYTE 2	DESCRIPTION OF ERROR
8	0xFF	Service not supported (invalid Service Code)
0x16	0xFF	Object does not exist (invalid Instance ID or Class ID)
0x20	1	Register base address is not in the proper form
0x20	2	Register quantity is not in the proper form
0x20	3	Register base address out of range for this CPU model
0x20	4	Register quantity out of range for this CPU model
0x20	5	Sum of register base and quantity are out of range for this CPU
0x20	6	Error communicating with PLC on GET_PLC_RGS service
0x20	7	Error communicating with PLC on SET_PLC_RGS service

The following General Error Codes are defined by DeviceNet.

Table 5.6 – General Error Codes		
Error Code (In Hex)	Name	Description of Error
00 – 01	Reserved	
02	Resource Unavailable	Resource needed for the object to perform the requested service were not available
03 – 07	Reserved	
08	Service Not Supported	The requested service was not implemented or not defined for this Object Class/Instance
09	Invalid Attribute Value	Invalid attribute data detected
0A	Reserved	
0B	Already in requested mode	The object is already in the requested mode or state requested by the service
0C	Object State Conflict	The object can not perform the requested service in it's current mode or state
0D	Reserved	
0E	Attribute Not Settable	A request to modify a non-modifiable attribute was received
0F	Privilege Violation	Permission/privilege check failed
10	Device State Conflict	The device's current mode or state prohibits the requested service
11	Reply Data Too Large	The data to be transmitted is large than the allocated response buffer
12	Reserved	
13	Not Enough Data	The service did not supply enough data to perform the requested service
14	Attribute Not Supported	the attribute specified in the request is not supported
15	Too Much Data	The serve supplied more data than was expected
16	Object Does not Exist	The specified object does not exist in the device
17	Reserved	
18	No stored attribute data	The attribute data of the object was not stored prior to the requested service
19	store operation failure	The attribute data of this object was not saved by the object
1A – 1E		Reserved by DeviceNet
1F	Vendor Specific error	A vendor-defined error has occurred. See the Additional Byte for further information
20	invalid parameter	A parameter associated with the service was determined to be not valid
21 – CF	Reserved	
D0 – FF	Object Class and Service Errors	Vendor-specific Object and Class errors.

