

1782-JDO
DeviceNet-to-Optomux Gateway
User's Manual



Western Reserve Controls, Inc.

Although every effort has been made to insure the accuracy of this document, all information is subject to change without notice. Western Reserve Controls, Inc. assumes no liability for any errors or omissions in this document or for direct, indirect, incidental or consequential damage resulting from the use of this document.

Rev 1.02
May 2004

Copyright © 2003-2004 WRC

Western Reserve Controls, Inc.

1485 Exeter Road
Akron OH 44306
330-733-6662 (Phone)
330-733-6663 (FAX)
sales@wrcakron.com (Email)
<http://www.wrcakron.com> (Web)

WRC is a trademark of Western Reserve Controls, Inc.
DeviceNet is a trademark of the Open DeviceNet Vendor Association ("ODVA").
Optomux is a trademark of Opto22.
All other trademarks are property of their respective companies.

TABLE OF CONTENTS

| | | |
|----------|--|-----------|
| 1 | OVERVIEW | 4 |
| 1.1 | FEATURES..... | 5 |
| 1.1 | OPTOMUX COMMAND SUMMARY..... | 6 |
| 1.2 | BASIC OPERATION..... | 6 |
| 1.2 | DEFAULT DEVICE CONFIGURATION..... | 7 |
| 1.3 | PRODUCT VERSION AND EDS..... | 7 |
| 2 | QUICK START | 8 |
| 2.1 | HOW TO INSTALL A SERIAL NETWORK..... | 8 |
| 2.2 | HOW TO READ DISCRETE INPUT DATA FROM THE JDO..... | 8 |
| 2.3 | HOW TO WRITE SERIAL OUTPUT DATA TO THE JDO..... | 10 |
| 2.4 | HOW TO CHANGE THE NODE ADDRESS..... | 11 |
| 2.5 | HOW TO CHANGE THE BAUD RATE..... | 11 |
| 3 | GENERAL SPECIFICATIONS | 12 |
| 4 | HARDWARE INSTALLATION AND SET-UP | 14 |
| 4.1 | OVERVIEW..... | 14 |
| 4.2 | LED OPERATION..... | 14 |
| 4.2.1 | <i>DeviceNet LED's</i> | 14 |
| 4.2.2 | <i>Serial Port LED's</i> | 15 |
| 4.3 | SERIAL PORT CONNECTOR..... | 16 |
| 4.4 | DEVICENET CONFIGURATION..... | 16 |
| 4.4.1 | <i>DeviceNet Network Termination</i> | 16 |
| 4.4.2 | <i>DeviceNet Connection Wiring</i> | 16 |
| 4.4.3 | <i>RS485 Network Termination (1782-JDO-2)</i> | 17 |
| 5 | SOFTWARE CONFIGURATION AND SET-UP | 18 |
| 5.1 | SETTING UP SERIAL COMMUNICATIONS..... | 20 |
| 5.1.1 | <i>Reading the Data Frame Format</i> | 20 |
| 5.1.2 | <i>Setting up the serial link baud rate</i> | 20 |
| 5.2 | SETTING UP DEVICENET COMMUNICATIONS..... | 20 |
| 5.2.1 | <i>Polled I/O</i> | 20 |
| 5.2.2 | <i>Reading the I/O Poll Response Data Size</i> | 20 |
| 5.2.3 | <i>Reading the I/O Poll Data Size</i> | 21 |
| 5.2.4 | <i>Setting up the DeviceNet Baudrate</i> | 21 |
| 5.2.5 | <i>Autobaud Operation</i> | 21 |
| 5.2.6 | <i>Setting up the Connection Timer (EPR)</i> | 21 |
| 6 | DEVICENET PROFILE, OBJECTS AND SERVICES | 22 |
| 6.1 | 1782-JDO DEVICENET PROFILE..... | 22 |
| 6.2 | IDENTITY OBJECT, CLASS 1..... | 22 |
| 6.3 | PARAMETER OBJECT, CLASS FHEX (15DEC)..... | 24 |
| 6.4 | COMMON DEVICENET SERVICES..... | 25 |
| 7 | ACCESSORIES AND OTHER WRC PRODUCTS | 26 |

LIST OF TABLES

| | |
|---|----|
| Table 1-1 I/O Message Types | 6 |
| Table 2-1 JDO Input Data Format (10 B1's)..... | 9 |
| Table 2-2 Input Status/Error Byte..... | 10 |
| Table 2-3..... | 10 |
| Table 2-4 DeviceNet Consume Assembly / Transmit Data Format | 11 |
| Table 2-5 Baud Rate Selection | 11 |
| Table 4-1 Module Status LED (labeled MS)..... | 15 |
| Table 4-2 Network Status LED (labeled NS)..... | 15 |
| Table 4-3 RS-232 / RS-485 Connector Signals | 16 |
| Table 4-4 Maximum Network Cable Lengths..... | 16 |
| Table 5-1 Configuration Parameter List | 18 |
| Table 5-2 Serial Baud Rates | 20 |
| Table 6-1 DeviceNet Objects | 22 |
| Table 6-2 Identity Object Class Attributes (Instance 0) | 22 |
| Table 6-3 Identity Object Instance Attributes (Instance 1) | 23 |
| Table 6-4 Identity Object Common Services | 23 |
| Table 6-5 Parameter Class Attributes (Instance 0)..... | 24 |
| Table 6-6 Parameter Instance Attributes (Instances 1-7) | 24 |
| Table 6-7 Parameter Common Services | 24 |
| Table 6-8 JDO Parameter Instances (Class F _{hex})..... | 24 |
| Table 7-1 WRC Replacements, Spare Parts and Other Products | 26 |

TABLE OF FIGURES

| | |
|---|----|
| Figure 1-1 1782-JDO | 5 |
| Figure 4-1 1782-JDO Outline Drawing..... | 14 |
| Figure 4-2 DeviceNet cable connector..... | 17 |

PRELIMINARY

1 Overview

WRC's 1782-JDO is a DeviceNet-to-RS485 serial link communications gateway and protocol converter that provides a DeviceNet interface to Optomux B1 Brainboards. The JDO allows the user to easily and conveniently connect and integrate existing B1 devices and their associated I/O into a DeviceNet system.

Using the 1782-JDO you may communicate with the connected B1's in the same fashion as the other DeviceNet products in the system. Data may be read/written using either I/O polling or explicit messaging. Typically real-time data is read and written as I/O by the DeviceNet Master via Polled I/O and parameters are read and written with the Explicit Messaging technique.

The JDO gateway has several pre-defined Optomux commands built that it uses to communicate with the B1 boards. As a user you do not need to know any details about the Optomux protocol – you simply read and write discrete I/O data in your DeviceNet Master's I/O tables. To get started you need to fill in the appropriate configuration Parameters in the JDO Parameter list to define your B1 Brainboard system and some general set-up attributes and then place the JDO in your DeviceNet Master's scan table. The JDO builds its own internal B1 scan list and starts communicating with the B1 boards. The JDO then translates discrete I/O data between the DeviceNet polled I/O data and the Optomux B1 protocol format.

You can connect up to 10 B1 brainboards to the JDO. The DeviceNet Master reads 3 bytes of data from the JDO (1 status and 2 input data bytes) and writes 2 bytes of discrete output data to the JDO for each brainboard.

The 1782-JDO is defined as a Communications Adapter device on the DeviceNet system. It has a 3-pin plug connector for connection to a RS485/422 converter that then connects to the to and from host connections of the port on a B1 and a 5-pin pluggable DeviceNet connector for connections to the DeviceNet network. The serial port is configurable to run up to 38.4 kbaud. The device does auto baud rate selection when it is powered up on the DeviceNet network. The 1782-JDO has one assigned DeviceNet address, which is set by a 6position DIPswitch on the unit. Other JDO parameters are software-configurable and are changed from their default values by third-party DeviceNet configuration tools. Each 1782-JDO has 2 standard green/red DeviceNet LED's for module status and network status and two green LED's to indicate RS485 transmit and receive activity. The RS485 link may be connected in a point-to-point fashion to a single device, or to multiple devices in the standard RS485 convention.

This manual applies to 1782-JDO software versions 1.08 and later.

PRELIMINARY



Figure 1-1 1782-JDO

1.1 Features

The 1782-JDO has the following features:

- Translates messages and data between DeviceNet and up to 10 Opto22 B1 devices
- ODVA Conformance tested to DeviceNet Spec 2.0
- Defined as a DeviceNet Communications Device Profile 12 (C_{hex})
- Autobaud operation
- Polled I/O and Explicit Messaging
- Software Configurable Parameters for serial port operation
- Software Configurable Parameters to define the B1 system
- DeviceNet address selection via DIP switches
- DIN rail mount
- Pluggable 5-pin DeviceNet connection
- Pluggable RS-485 3-pin connection
- 2 standard DeviceNet module and network status LED's
- Serial transmit and receive green LEDs
- Powered from DeviceNet 11-25 Vdc network power
- Serial port baud rate up to 38.4k baud

PRELIMINARY

1.1 Optomux Command Summary

The JDO supports the following B1 commands:

- Power-up Clear (command A)
- Reset (command B)
- Enhanced Digital Watchdog (command m)
- Configure Positions (command G)
- Read Module Configuration (command j)
- Write Outputs (command J)
- Read Status (command M)

1.2 Basic Operation

The JDO operates as the DeviceNet front-end to the Optomux B1 brainboards. The DeviceNet Master can receive and send data to and from the 1782-JDO via the methods described in this section. It formats and sends the data to the B1s and accepts responses from the B1, which are reformatted and passed back to the DeviceNet system as required. In short, the JDO behaves like an Optomux scanner that is connected to DeviceNet.

The JDO has one DeviceNet address. All DeviceNet messages to the JDO are sent to this address. All DeviceNet messages to and from the B1's are sent to the JDO DeviceNet assembly objects using poll commands.

The JDO Parameter Object allows you to define the specific operation of each JDO and the attached B1's. These parameters include all the set-up required for the serial comm. link.

The following chart defines the various messaging methods used for data types at your serial device and a brief explanation follows.

Table 1-1 I/O Message Types

| Typical Data | Polled | Cyclic | Bit-Strobe | Change-of-State | Explicit Message |
|--------------|--------|--------|------------|-----------------|------------------|
| Commands | 0 | | | | |
| Status | 0 | | | | |
| Parameters | | | | | 0 |

The general sequence of events and operation is explained below:

1. Configure the JDO for the number of brainboards (type B1) to be connected and the input/output definition of each.
2. Each B1 has 9 associated parameters, including its Optomux address, input/output

PRELIMINARY

configuration, watchdog operation, etc. Set these parameters for each one.

3. Set up the rest of the parameters, as appropriate.
4. **Note: Once the JDO is configured, it must be reset to complete the set up and configuration of the B1's.**
5. Determine the DeviceNet I/O command and response data sizes – 3 bytes input and 2 bytes output data for each B1 – by reading the JDO Produce and Consume Sizes in read-only Parameters 5 and 6.
6. Enter the JDO DeviceNet I/O sizes into your Master scanner and map its I/O into the scanner's I/O data table.
7. Once the DeviceNet scanner starts scanning the network, the JDO will start a sequence of continuous reads and writes of each connected B1 to receive and transmit its output and input data.
8. The JDO will continuously update its DeviceNet produce buffer asynchronously with B1 input data and provide updated information to the DeviceNet scanner via the Polled I/O Response technique.
 - The input data will be placed in the input field in the order as the B1's are listed in the Parameter list of the JDO.
9. Once the control system is put in run mode, the JDO will take the output data received from the DeviceNet Master in the Poll Command and sent the output data to each B1.
 - The output data will be read from the output field (consume data) in the order as the B1's are listed in the Parameter list of the JDO.

1.2 Default Device Configuration

The 1782-JDO DeviceNet address is read from the switches and is set to 63 and the baud rate is set to Autobaud at the factory. All other parameters are software settable. The default settings for the 1782-JDO are provided in the discussion of the Parameter Object and Table 5-1.

1.3 Product version and EDS

This manual applies to 1782-JDO versions 1.08 and higher. An EDS (Electronic Data Sheet) for the 1782-JDO, is shipped with your device or is available on WRC's web site:

<http://www.wrcakron.com/>.

PRELIMINARY

2 Quick Start

To quickly install your 1782-JDO in your DeviceNet system, follow the instructions below. *For more details, see Section 4.*

1. Connect your DeviceNet network cable to a 5pin female (Phoenix-type) plug according to DeviceNet cable wiring specifications
2. Make sure that the DeviceNet network is terminated properly.
3. The JDO Node Address (MacID) is set to 63 at the factory. Make sure no other device on the network is set to 63, or change the JDO address to one that is not currently used (see Section 2.4 below).
4. The JDO baud rate is set to Autobaud operation at the factory. See Section 2.5 below to change it to a fixed baud rate if desired.
5. Make sure that there is power on the DeviceNet network and plug the cable into the 1782-JDO.
6. The 1782-JDO will undergo its initialization sequence, flashing both LED's red and green. After approximately 5 seconds, the Module Status LED (labeled "MS") will flash green. The Network Status LED (labeled "NS") will remain off. This condition occurs while the JDO is attempting to synchronize to the network baudrate (autobaud).
7. The Module Status LED ("MS") will go on solid after the Device successfully determines the network baudrate. This requires devices on the network attempting to communicate with each other. The Network Status LED (labeled "NS") will begin to flash green. If it turns solid red, check for a duplicate MacID on the network. If it remains off, make sure that there are other devices ***trying to communicate*** on the network.
8. Once the Master recognizes the unit on the link and allocates the connection (initiates communications), The Network Status LED will be solid green. The device is now being actively scanned.
9. The 1782-JDO is now operating on the network.

2.1 How to Install a Serial Network

1. The communication between your serial device(s) and the 1782-JDO is an RS-485 2-wire differential network. The B1's are 4-wire RS-422 devices.
2. Connect one end of the cable to the JDO using the 3point terminal plug provided. Note the terminal markings on the JDO case.
3. Connect the JDO SIG+ signal wire to the TH+ and FH+ terminals on the first B1.
4. Connect the JDO SIG- signal wire to the TH- and FH- terminals on the first B1.
5. Turn on power to the serial device and the JDO.

Note: Opto22 and WRC recommend an external converter to make the interconnection between the JDO and a Brainboard a RS-422 connection.

2.2 How to Read Discrete Input Data from the JDO

1. Set up the receive size of your connection to equal the **(Max Number of B1 Devices * 3 (+1 if the Max Number is odd))** (set in Parameter 4) in your Master's scan table. This

PRELIMINARY

value is calculated by the JDO using the number of B1 devices you define in Parameter 4 and can be read from Parameter 5 in the Parameter Object and the EDS file. The default value should be 4.

2. Map the data from your IO response to your scanners memory map.
 - i. The first byte indicates the status byte from the B1.
 - ii. The second byte contains the DI's for channels 0-7.
 - iii. The third byte contains the DI's for channels 8-15.

3. Direct the master to begin polling the JDO. The JDO will continuously issue Read Status commands to the B1 and update the DeviceNet Poll Response (input data) with the input data from the B1.

Table 2-1 JDO Input Data Format (10 B1's)

| BYTE | B1 # | MEANING |
|---------|------------------|----------------------|
| Byte 0 | 1 st | data byte, DI's 0-7 |
| Byte 1 | 1 st | data byte, DI's 8-15 |
| Byte 2 | 2 nd | data byte, DI's 0-7 |
| Byte 3 | 2 nd | data byte, DI's 8-15 |
| Byte 4 | 3 rd | data byte, DI's 0-7 |
| Byte 5 | 3 rd | data byte, DI's 8-15 |
| | | |
| Byte 18 | 10 th | data byte, DI's 0-7 |
| Byte 19 | 10 th | data byte, DI's 8-15 |
| Byte 20 | 1 st | status byte |
| Byte 21 | 2 nd | status byte |
| Byte 25 | 3 rd | status byte |
| | | |
| Byte 29 | 10 th | status byte |

PRELIMINARY

The status/error byte represents the status of the communication command from the JDO to the B1. This byte is defined as follows:

Table 2-2 **Input Status/Error Byte**

Table 2-3

| Byte Value (decimal) | Byte Value (hex) | Meaning |
|-------------------------|---------------------|---|
| 0 | 0 | No Error |
| 1 | 1 | Power-up Clear Command Expected* |
| 2 | 2 | Undefined Command* |
| 3 | 3 | TX Checksum Error to B1 |
| 4 | 4 | Input Buffer Overrun* |
| 5 | 5 | Non-printable ASCII Character Received* |
| 6 | 6 | Data Field Error* |
| 7 | 7 | Communications Link Watchdog Time-out Error |
| 8 | 8 | Specified Limits Invalid |
| 10 | A | RX Checksum Error from B1 |
| 15 | F | Optomux unit did not respond |

* These are defined Optomux errors (Optomux error # + 1) that should never be seen with a properly operating JDO.

4. Set up the JDO in your DeviceNet Master's scan list.
5. Map your DeviceNet I/O between the JDO and the Master's I/O image table or other suitable file. (The JDO defaults to one B1 unit, so the JDO will send 3 bytes of data and receive 2 from the Master.)

2.3 How to Write Serial Output Data to the JDO

1. Set up the receive size of your connection to equal the **(Max Number of B1 Devices * 2)** in your Master's scan table. This value is calculated by the JDO using the number of B1 devices you define in Parameter 4 and can be read from Parameter 6 in the Parameter Object and the EDS file. The default value should be 2.
2. Map the data for your IO command to your scanners memory map.
 - i. The second byte contains the DI's for channels 0-7.
 - ii. The third byte contains the DI's for channels 8-15.
3. Direct the master to begin polling the JDO. The JDO will continuously issue Read Status commands to the B1 and update the DeviceNet Poll Response (input data) with the input data from the B1.

PRELIMINARY

Table 2-4 **DeviceNet Consume Assembly / Transmit Data Format**

| BYTE | MEANING |
|---------|--|
| Byte 0 | 1 st B1 data byte, DO's 0-7 |
| Byte 1 | 1 st B1 data byte, DO's 8-15 |
| Byte 2 | 2 nd B1 data byte, DO's 0-7 |
| Byte 3 | 2 nd B1 data byte, DO's 8-15 |
| | |
| Byte 19 | 10 th B1 data byte, DO's 8-15 |

4. Begin scanning the JDO. Enter data that you want to send on the serial link. On receipt of a change to the record number, the JDO will transmit the data. If any data is in the receive buffer, the received data will be returned as a poll response. (If the sequence number is the same as the previous response, the JDO will not re-send the string.)

2.4 How to Change the Node Address

1. Set the 6-position DIPswitch to the binary number representing the desired Node Address, 0-63. (Note: Address 0 is often reserved for a Master device.)
2. Power cycle the unit by unplugging and reconnecting the DeviceNet cable.

NOTE: The new address will not become effective until the unit is power cycled or a Reset command is received from the Master.

2.5 How to Change the Baud Rate

The Baud Rate is set to Autobaud at the factory. The baudrate can be changed through your configuration tool in its normal manner to any baudrate except Autobaud. This is due to the fact that Autobaud is not provided for in the DeviceNet specification. If you need to set the baudrate to Autobaud, you must do it via the parameter object. The definition is included in the EDS file for easy configuration. Just use your configuration tool to access the device parameters. The baudrate is parameter 7. Select the proper baudrate and upload the parameter to the device. If your configuration tool does not support EDS parameter configuration, you will have to perform the operation manually. To do this, set the parameter Class 15 (F_{hex}), instance 7, attribute 1 to the value in Table 2-5.

NOTE: If you change the baudrate, the new baudrate will not become effective until the unit is power cycled or a reset command is received by the Identity Object (Class 1, Instance 0, reset).

Table 2-5 **Baud Rate Selection**

| Baud Rate Value | Baud Rate |
|-----------------|-----------|
| 0 | 125k |
| 1 | 250k |
| 2 | 500k |
| 3 | Autobaud |

PRELIMINARY**3 General Specifications**

| | |
|----------------------------------|---|
| Product: | 1782-JDO Device-Optomux B1 Gateway |
| Description: | Communications gateway between one or more Optomux B1 device(s) on an RS485 interface and a DeviceNet network. |
| Device Type: | Communications Adapter, C _{hex} , (12) |
| Device Profile: | Identity Object Message Router Object DeviceNet Object Connection Object Parameter Object Assembly Objects (2) |
| Product Revision: | 1.01 |
| DeviceNet Conformance: | Designed to conform to the ODVA DeviceNet Specification Volume I and II, Version 2.0. |
| DeviceNet Communications: | Predefined Master/Slave Connection Set, Group 2 Only Server |
| DeviceNet: | Baud rate selection: Autobaud operation (default) Fixed baud (software selectable) – 125k, 250k and 500k baud |
| | Address selection: Address number 0 to 63, switch selectable (default = 63) |
| | Cable Connection: JDO: 5-pin pluggable header (male) Phoenix Contact MSTBA 2.5/5-G-5.08/AU or equivalent DeviceNet Cable: 5-contact plug (female contacts) Phoenix Contact MSTB 2.5/5-ST-5.08/AU or equivalent (included) |
| | Status Indicators: Module Status: green/red bi-color LED Network Status: green/red bi-color LED |
| Serial port: | Baud rate: 1200, 2400, 4800, 9600, 19.2k, 38.4k baud (software selectable) |
| | Character Framing: 8-N-1 |
| | Cable Connection: JDO: 3-pin pluggable header (male) Phoenix Contact MSTBA 2.5/3-G-5.08/AU or equivalent Serial Cable: 3-contact plug (female contacts) Phoenix Contact MSTB 2.5/3-ST-5.08/AU or equivalent (included) |
| | Status Indicators: Transmit Active: green LED Receive Active: green LED |
| Network Isolation: | (Optional 2500V RS232) |

PRELIMINARY

| | |
|------------------------|--|
| Max Power: | 1.75 watts: 160 mA @ 11 Vdc – 70 mA @ 25 Vdc unregulated power supply |
| Mounting: | DIN rail mount, EN 50022 |
| Size: | <ul style="list-style-type: none">●Depth: 3.54" (90 mm)●Width: 0.98" (25 mm)●Height: 3.11" (79 mm) |
| Operating Temp: | 0-60 °C |
| Humidity: | 0-95% RH, non-condensing |

PRELIMINARY

4 Hardware Installation and Set-Up

4.1 Overview

The JDO is mounted on an EN50022 DIN rail.

The JDO contains two LED's to indicate the status of the device and the status of the network. The device can be connected to the main DeviceNet trunk line or to a drop line via a 5-pin female plug-style connector. It also has two green LED's to indicate the presence of activity on the RS-485 transmit and receive lines.

All power for the JDO is derived from the DeviceNet power.

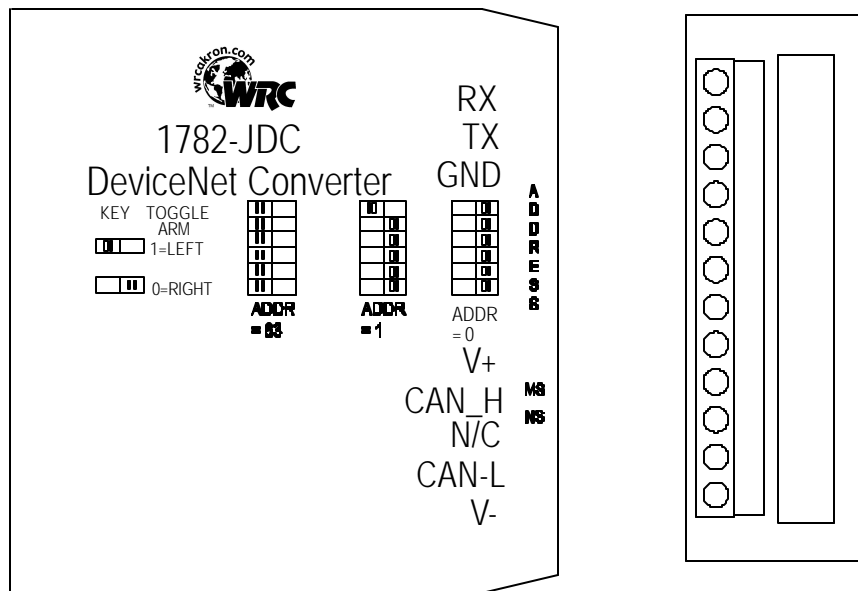


Figure 4-1 1782-JDO Outline Drawing

4.2 LED Operation

4.2.1 DeviceNet LED's

The JDO has two LED's that provide visual status information to the user about the product and the DeviceNet network. See Tables 51 and 52 that follow below for how to interpret LED status indications.

PRELIMINARYTable 4-1 **Module Status LED (labeled MS)**

| LED State | Module Status | Meaning |
|--------------------|---------------------|---|
| OFF | No Power | There is no power through DeviceNet. |
| Green | Device Operational | JDO is operating normally. |
| Flashing Green | Device in Standby | JDO needs commissioning (e.g. attempting Autobaud). |
| Flashing Red | Minor Fault | Recoverable fault. |
| Red | Unrecoverable Fault | JDO may need replaced. |
| Flashing Red/Green | Device Self-Testing | JDO is in self-test mode. |

Table 4-2 **Network Status LED (labeled NS)**

| LED State | Network Status | Meaning |
|----------------|------------------------|--|
| OFF | No Power / Not on-line | JDO has no power or has not completed the Dup_MAC_ID test. |
| Flashing Green | On-line, not connected | JDO is on-line but is not allocated to a Master. |
| Green | On-line | JDO is operating normally. |
| Flashing Red | Connection time-out | One or more I/O connections are timed out. |
| Red | Critical link failure | JDO has detected an error that makes it incapable of communicating on the link. (Bus off or Duplicate MAC ID). |

4.2.2 Serial Port LED's

The JDO also has two (2) RS-485 activity LED's: one for transmit (TX) and one for receive (RX). Each of these will illuminate when there is data communications active on the respective data lines.

PRELIMINARY

4.3 Serial Port Connector

The ASCII devices are connected to the JDO via a 3-wire communications cable. See your ASCII device's User Manual for details on the proper connections.

The RX and TX designators are referenced with respect to the JDO.

Table 4-3 RS-232 / RS-485 Connector Signals

| Pin # | Designator | RS232 Signal | RS485 Signal |
|-------|------------|--------------|--------------|
| 3 | RX | Receive | Signal - |
| 2 | TX | Transmit | Signal + |
| 1 | GND | Ground | Common |

4.4 DeviceNet Configuration

DeviceNet specifications provide for a maximum network distances for the main trunk line and drop lines, depending upon the baud rate used on the network. See Table 4-4

Table 4-4 Maximum Network Cable Lengths

| Baud Rate | Trunk Line Length | | Drop Length | | | |
|-----------|-------------------|---------|-------------|-------|------------|---------|
| | Maximum Distance | | Maximum | | Cumulative | |
| | Meters | Feet | Meters | Feet | Meters | Feet |
| 125k baud | 500 m | 1640 ft | 6 m | 20 ft | 156 m | 512 ft. |
| 250k baud | 250 m | 820 ft | 6 m | 20 ft | 78 m | 256 ft. |
| 500k baud | 100 m | 328 ft | 6 m | 20 ft | 39 m | 128 ft. |

4.4.1 DeviceNet Network Termination

A DeviceNet system **must be terminated at each end of the trunk line**. The host controller and the **last** JDO or other DeviceNet device on the network must always be terminated to eliminate reflections, even if only two nodes are present. The DeviceNet specifications for the terminating resistor are:

- 121 ohm
- 1% metal film
- 1/4 Watt

An appropriate terminating resistor, WRC part number RM121DN, may be purchased from WRC.

IMPORTANT: Per the DeviceNet spec -- do not terminate devices on drop lines.

4.4.2 DeviceNet Connection Wiring

The JDO uses a 5-pin plug-style DeviceNet connector, which has male pins.

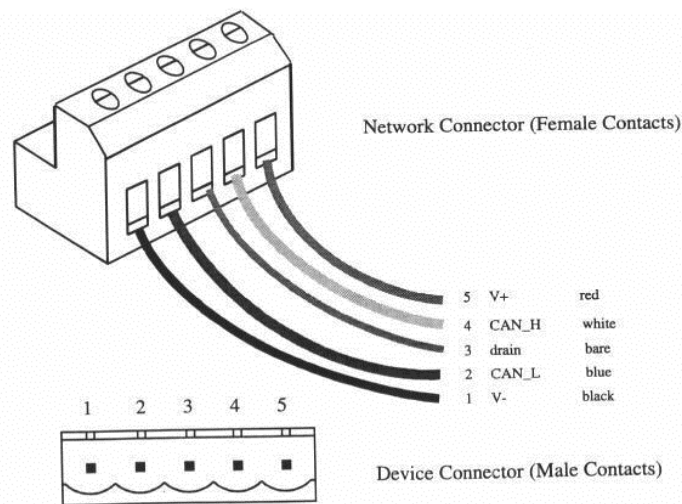
PRELIMINARY

Figure 4-2 DeviceNet cable connector

4.4.3 RS485 Network Termination (1782-JDO-2)

It is common practice to terminate an RS485 network system **at each end of the serial cable** to eliminate electrical reflections. An appropriate value for the terminating resistor typically is:

- 100-120 ohm, depending upon the characteristic impedance of the cable used
 - 121 ohms (same value as the DeviceNet terminator) is a standard resistor value that may be used in many cases
- 1% metal film
- 1/4 Watt

Correct cable termination is the responsibility of the user. The 1782-JDO-2 does not include an internal terminating resistor.

PRELIMINARY**5 SOFTWARE Configuration and Set-Up**

The 1782-JDO is an easy device to set up and configure. Using features like the EDS sheets for configuration can expedite the process if you use a network configuration tool that supports them. They provide a graphical interface to the device's parameters and allow the addition of helpful text descriptions in setting up your device. The current EDS file is available on our website, www.wrcakron.com. If your configuration tool does not support the EDS device profiles, the set up of a DeviceNet device requires a little more understanding of DeviceNet and its operation. This section is designed to fully describe the features of the 1782-JDO and to help you set them up. If you have problems setting up a device, we are available to help you. We can be reached via phone at (330) 733-6662, or via the Internet at support@wrcakron.com.

The Parameters are defined in this section.

Table 5-1 Configuration Parameter List

| Parameter | Param. Instance | Access | Description | Parameter Choices | Default Setting | Default Value | Data Type |
|--------------------------|-----------------|---------|---|--|-----------------|---------------|-----------|
| Serial Character Format | 1 | Get | Character framing | Optomux brainboards use only 8N1 | 8N1 | 3 | USINT |
| Serial Baud Rate | 2 | Get/Set | RS232/ RS485 communications speed | 0 = 9600 1 = 1200 2 = 2400 3 = 4800 4 = 19.2k 5 = 38.4k | 9600 baud | 0 | USINT |
| Device Net Baud Rate | 3 | Get/Set | The baud Rate of the device net device. The baud rate can be set here and will take effect once the JDO has been reset. | 0 = 125 Kbaud 1 = 250 Kbaud 2 = 500 Kbaud 3 = Autobaud | Autobaud | 3 | USINT |
| Max Number of B1 Units | 4 | Get/Set | Maximum number of B1 units the JDO expects to read and write via the serial port | 1 – 20 | 1 unit | 1 | USINT |
| Number of Input Bytes | 5 | Get | Number of bytes the JDO will send to the DeviceNet Master | 4-60 | One B1 | 4 | USINT |
| Number of Output Bytes | 6 | Get | Number of bytes the JDO will receive from the DeviceNet Master | 2-40 | One B1 | 2 | USINT |
| Message Response Timeout | 7 | Get/Set | Timeout value for Max time the JDO will wait for each unit on the Optomux network. | 1-65535 ms | 100 | 100 | UINT |
| Estimated Scan Time | 8 | Get | Estimated time required to complete a full scan of Optomux network | None | none | none | UINT |
| Actual Scan Time | 9 | Get | Actual time required to complete a full scan of Optomux network | None | none | none | UINT |
| BB#1 Address | 10 | Get/Set | Optomux address for 1 st B1 | 1-255 | 1 | 1 | USINT |
| BB#1 Type | 11 | Get/Set | Optomux type | 0-1 | B1 | 0 | USINT |
| BB#1 I/O | 12 | Get/Set | Input/output | 0x0000-0xFFFF | All inputs | 0x0000 | UINT |

PRELIMINARY

| Parameter | Param. Instance | Access | Description | Parameter Choices | Default Setting | Default Value | Data Type |
|------------------------------------|-----------------|---------|--|--|-----------------|---------------|-----------|
| Configuration | | | configuration | 1 bit per I/O point 0 = input 1 = output | | | |
| BB#1 Watchdog Action | 13 | Get/Set | Action taken on output if no serial com with Optomux | 0x0000-0xFFFF | 0x0000 | 0x0000 | UINT |
| BB#1 Watchdog Delay | 14 | Get/Set | Time until action is taken on output if no serial com with Optomux in 10ms ticks | 0-65535 in 10ms ticks 0 = Disabled | Disabled | 0 | USINT |
| BB#1 DN Output Fault Action | 15 | Get/Set | Output actions if no DeviceNet communications | 0x0000-0xFFFF 1 bit per output: 0 = use fault value 1 = hold last state | Use Fault Value | 0x0000 | UINT |
| BB#1 DN Output Fault Value | 16 | Get/Set | Output actions if no DeviceNet communications | 0x0000-0xFFFF 1 bit per output: 0 = turn OFF 1 = turn ON | Turn OFF | 0x0000 | UINT |
| BB#1 DN Output Idle Action | 17 | Get/Set | Output actions when DeviceNet in Idle mode | 0x0000-0xFFFF 1 bit per output: 0 = use Idle value 1 = hold last state | Use Idle Value | 0x0000 | UINT |
| BB#1 DN Output Idle Value | 18 | Get/Set | Output actions when DeviceNet in Idle mode | 0x0000-0xFFFF 1 bit per output: 0 = turn OFF 1 = turn ON | Turn OFF | 0x0000 | UINT |

Note: Parameters 10 through 18 are repeated 9 more times for the other 9 brainboards that can be on the JDO's local network.

Example: Parameters 19,28,37,46,55,64,73,82,and 91 are the same as parameter 10 above.

Parameter 19 is for brainboard #2. 28 is for brainboard #3, etc.

PRELIMINARY

5.1 Setting up Serial Communications

5.1.1 Reading the Data Frame Format

The data frame format parameter defines the size of the data frame that is transmitted and received over the serial link. This parameter defines the number of data bits per character (7 or 8), the parity (odd, even or none) and the number of stop bits (1 or 2). Optomux brainboards support 8N1 only. This parameter provides an affirmation of this value in the JDO.

5.1.2 Setting up the serial link baud rate

The standard baudrates are supported by the JDO, starting at 1200 baud and going to 38400 baud. The baudrate is configured at parameter 2 of the parameter object. The valid options are:

Table 5-2 Serial Baud Rates

| Value | Baud Rate |
|-------|-----------|
| 0 | 9600 |
| 1 | 1200 |
| 2 | 2400 |
| 3 | 4800 |
| 4 | 19200 |
| 5 | 38400 |

5.2 Setting up DeviceNet Communications

The 1782-JDO supports Polled I/O only.

5.2.1 Polled I/O

The polled connection is the only manner in which you can send serial output data to the JDO and, therefore, to your ASCII device. The DeviceNet Master initiates the polled connection by sending Poll Command data to the JDO for the B1's and the JDO responds with data from the B1's.

5.2.2 Reading the I/O Poll Response Data Size

This is the number of bytes that the JDO will send to the DeviceNet Master from the B1(s) and is determined by:

$$\text{produce_size} = \text{number_of_B1s} * 3 (+1 \text{ if the number of B1's is odd})$$

This number is calculated by the JDO. The valid settings for this value are 4-60. The default value is 4. This produce_size is used to set up the Poll Receive size at your Master device. These values can be retrieved by using the standard get service on class 15 (F_{hex}), instance 5, attribute 1 (See Table 5-1).

Note: The number of bytes in the response message will always be even, due to padding which takes place when the number of brainboards is odd. This is due to the fact that PLCs are 16 bit or 32 bit word length as formatted in memory.

PRELIMINARY

5.2.3 Reading the I/O Poll Data Size

This is the number of bytes that the JDO will receive from the DeviceNet Master and is determined by:

$$\text{consume_size} = \text{number_of_B1s} * 2$$

This number is calculated by the JDO. The valid settings for this value are 2-40. The default value is 2. This consume_size is used to set up the Poll transmit size at your Master device. These values can be retrieved by using the standard get service on class 15 (F_{hex}), instance 6, attribute 1 (See Table 5-1).

Note: There are always 2 bytes allocated for output data for each B1, irregardless of how many or whether any channels are configured as outputs.

5.2.4 Setting up the DeviceNet Baudrate

If you wish to change the baudrate to a fixed speed, you can set it in two different places. The first is the standard location for the DeviceNet baud rate, which is in the DeviceNet object. This is where most configuration tools will look to change the baudrate. This is fine on normal DeviceNet systems, however the JDO provides some extended DeviceNet functionality above normal systems to allow the Autobaud functionality. This extension is not defined by ODVA thus cannot be attached to any of the standard DeviceNet objects to ensure compatibility. You can still use the baudrate in the normal manner; it will just not allow the Autobaud setting.

This value can be set and retrieved by using the standard set and get services on class 15 (F_{hex}), instance 3, attribute 1. (See Table 5-1).

5.2.5 Autobaud Operation

Autobaud is the mechanism that allows the DeviceNet device, in this case the JDO, to automatically determine the baudrate that is operational on the network to which the JDO is connected and adjust its DeviceNet speed to match. The JDO is shipped with Autobaud as its default baudrate.

5.2.6 Setting up the Connection Timer (EPR)

EPR stands for Expected Packet Rate. This is the value that the JDO sets the connection timer to for the cyclic and polled connection. This is also the value it uses in the connections to calculate the time the device should wait before signaling a timeout. If you have a scanner or scanning software, you must configure it with the EPR that you want the JDO to be scanned with. The **scanner will then configure the EPR** in the JDO at the beginning of communications. Consult your scanner's manuals on how to configure the EPR (the EPR is sometimes referred to as the "scan rate").

Note: If you need to set up the EPR, it can be done manually by performing a set (Service 10_{hex}) on the connection class (Class 5_{hex}) attribute 9. The polled connection uses instance 2. This must be done after allocating the connection.

PRELIMINARY

6 DeviceNet Profile, Objects and Services

6.1 1782-JDO DeviceNet Profile

This section describes the DeviceNet Objects present in the JDO. The JDO conforms to a Type 12, Communications Adapter Device.

Table 6-1 DeviceNet Objects

| Object | DeviceNet Object Class | # of Instances |
|----------------|------------------------|-------------------------------|
| Identity | 1 | 1 |
| Message Router | 2 | 1 |
| DeviceNet | 3 | 1 |
| Connection | 5 | 2 (Explicit Msg., Polled I/O) |
| Parameter | 15 (E _{hex}) | 10 |

6.2 Identity Object, Class 1

Instances 0 and 1 exist in the 1782-JDO.

Table 6-2 Identity Object Class Attributes (Instance 0)

| Attribute ID | Access Rule | Name | DeviceNet Data Type | Description of Attribute | Value |
|--------------|-------------|-----------------------------|---------------------|--|-------|
| 1 | Get | Revision | UINT | Revision of this object | 1 |
| 2 | Get | Max. Object Instance | UINT | Maximum instance number of an object currently | 1 |
| 6 | Get | Max. Class Attribute ID | UINT | Attribute ID number of the last class attribute of the class definition implemented in the device | 7 |
| 7 | Get | Max. Instance Attributes ID | UINT | Attribute ID number of the last instance attribute of the class definition implemented in the device | 1 |

PRELIMINARY

Table 6-3 Identity Object Instance Attributes (Instance 1)

| Attribute ID | Access Rule | Name | DeviceNet Data Type | Description of Attribute | Value |
|--------------|-------------|--------------------|---------------------|---|--------------------------|
| 1 | Get | Vendor | UINT | ODVA Vendor Number for this product | 9 = WRC |
| 2 | Get | Device Type | UINT | ODVA Communications Device Type | 12 = Comm. Adapter |
| 3 | Set | Product Code | UINT | WRC Unique Product Code Number | 701 = 2bd _{hex} |
| 4 | Get | Revision | STRUCT of: | Revision of this device | |
| | | Major Revision | USINT | | >=5 |
| | | Minor Revision | USINT | | >=1 |
| 5 | Get | Status | WORD | Summary status of device | |
| 6 | Get | Serial Number | UDINT | WRC Unique Device Serial Number | |
| 7 | Get | Product Name | SHORT_STRING | ASCII Name of product | 1782-JDO |
| 10 | Get/Set | Heartbeat Interval | USINT | The interval in second that the device generates a heartbeat message. A value of 0 disables heartbeat generation. | 0 |

Table 6-4 Identity Object Common Services

| Service Code | Class | Instance | Service Name | Description of Service |
|-------------------|-------|----------|----------------------|--|
| 05 _{hex} | Yes | Yes | Reset | Invokes the Reset Service for the device. |
| 0E _{hex} | Yes | Yes | Get_Attribute_Single | Returns the contents of the specified attribute. |
| 10 _{hex} | No | Yes | Set_Attribute_Single | Modifies an attribute value. |

PRELIMINARY

6.3 Parameter Object, Class F_{hex} (15_{dec})

There are many configurable data parameters associated with your JDO. The JDO uses a Parameter Object (a collection of these parameters) to assist you in reading and changing configurable data.

Following are the Class Attributes, Instance Attributes and Services that are supported by the JDO for the Parameter Object.

Table 6-5 Parameter Class Attributes (Instance 0)

| Attribute ID | Access Rule | Name | DeviceNet Data Type | Description of Attribute | Value |
|--------------|-------------|----------------------------|---------------------|---|---|
| 1 | Get | Revision | UINT | Revision of this object. | 1 |
| 2 | Get | Max. Instance | UINT | Maximum instance number of the Parameter object | 9 |
| 8 | Get | Parameter class descriptor | WORD | Bits that describe parameters. | 9 (supports parameter instances, params are stored in non-volatile memory) |

Table 6-6 Parameter Instance Attributes (Instances 1-7)

| Attribute ID | Access Rule | Name | DeviceNet Data Type | Description of Attribute | Value |
|--------------|-------------|-----------------|---|--|-----------------|
| 1 | Set | Parameter Value | <i>data type</i> specified in Descriptor Data Type and Data Size. | Actual value of parameter. It can be read from or written to. This attribute is read-only if bit 4 of Attribute 4 is TRUE. | |
| 2 | Set | Link Path Size | USINT | Size of link path. If this is 0, then no link is specified. | Number of bytes |
| 3 | Set | Link Path | ARRAY of DeviceNet path: | DeviceNet path to the object from where this parameter's value is retrieved. | |
| 4 | Get | Descriptor | WORD | Description of parameter. | |
| 5 | Get | Data Type | USINT | Data type code. | |
| 6 | Get | Data Size | USINT | Number of bytes in Parameter Value | |

Table 6-7 Parameter Common Services

| Service Code | Class | Instance | Service Name | Description of Service |
|-------------------|-------|----------|----------------------|---|
| 05 _{hex} | Yes | N/A | Reset | Resets all parameters to "out-of-the-box" values. |
| 0E _{hex} | Yes | Yes | Get_Attribute_Single | Returns the contents of the specified attribute. |
| 10 _{hex} | Yes | Yes | Set Attribute Single | Modifies an attribute value. |

Table 6-8 JDO Parameter Instances (Class F_{hex})

For this table See Table 5-1

PRELIMINARY

6.4 Common DeviceNet services

The common DeviceNet services are available through many of the common configuration tools. However, each configuration tool may implement these differently. It may be easier if you use this section in conjunction with any files or documentation that was included with your DeviceNet configuration tools while going over this section.

DeviceNet is divided into logical functional blocks called objects, which provide services that allow for control over the hardware and routines that those objects contain. To allow for multiple similar functions, the objects are built of multiple instances that the services of the objects act upon. A class service acts upon the entire object, allowing one service to be enacted on all of the instances. This saves time, effort and network bandwidth.

The common services are a common set of services that have been provided in most or all of the objects to allow for common functionality in creating, deleting, getting, setting and resetting the variables of the different classes and instances. We will describe two of the services here: get and set.

The get and set services have a common format for specifying what object, instance, attribute and service that the command is specifying. In order of first to last, DeviceNet specifies service, class, instance, attribute and data. The data is always little-endian (low byte precedes high order byte), and the others are all one byte in length on the JDO. Note that the get service has no data.

The get service gets data from an attribute of a class or class instance. The service number of this request is 14 (E_{hex}). The class instance and attribute are all defined by which variable you want to get. The response from a get command takes on the form: service, value. The value will be little-endian and can be of variable length and bounds based on the definition of the attribute. The service will be reported as the get service with the highest bit set to indicate a response.

The set service sets data from an attribute of a class or class instance. The service number of this request is 16 (10_{hex}). The class instance and attribute are all defined by which variable you want to set. The value is little endian, and the size is defined by the attribute that you are setting. The response from a set command only echoes the service. The service will be reported as the set service with the highest bit set to indicate a response.

An error response will have the service set to 94_{hex} . This response will be followed by a two-byte error code, defining the type of fault. For a detailed list of error codes, connect to the ODVA web site at www.odva.org.

PRELIMINARY**7 Accessories and Other WRC Products**

The following components can be used with an Ajax for replacements or spare parts, or as complementary devices as a part of your DeviceNet or other CAN-Bus system.

Table 7-1 **WRC Replacements, Spare Parts and Other Products**

| Part | WRC Part Number |
|--|------------------------|
| DIN rail | WRC 50022 |
| Terminating resistor, axial lead | RM121DN |
| Discrete I/O block – 4 channels | 1782-JDB4 |
| Discrete I/O block – 8 channels | 1781-JDB8 |
| Analog Input block – 4 channels, 10-bit | 1782-JDA4 |
| Analog I/O block – 8 channels, 12-bit | 1782-JDA8 |
| DeviceNet to Serial I/O Gateway | 1782-JDC |
| DeviceNet to ModBus Gateway | 1782-JDM |
| DeviceNet to Optomux B1 Gateway | 1782-JDO |
| DeviceNet to Pamux Gateway | 1782-JDP |
| Discrete I/O block – 24 channels | WRC1-JDB24 |
| Discrete I/O block – 48 channels | WRC1-JDB48 |
| Discrete I/O, Analog Input block – 24 DIO, 32 AI | WRC1-JDA/24 |
| Discrete I/O, Analog Input block – 48 DIO, 32 AI | WRC1-JDA/48 |
| Analog I/O block - 32 channels | WRC1-JDAIO |
| Discrete and Analog I/O block – 24 DIO, 32 AIO | WRC1-JDAIO/24 |
| Discrete and Analog I/O block – 48IO, 32 AIO | WRC1-JDAIO/48 |
| Discrete I/O block – 8 DIs, 8 DOs, 4 AIs | W5-JDB16 |
| DeviceNet-to-ASCII Gateway – 4 channels | W5-JDC4 |
| DeviceNet, CANopen Extender, DIN mount | WRC-CANX-DIN-DN |
| SDS Extender, DIN mount | WRC-CANX-DIN-SD |
| DeviceNet, CANopen Extender, DIN mount | WRC-CANX-DIN-C7 |
| DeviceNet, CANopen Extender, NEMA box | WRC-CANX-NEM-AU |
| DeviceNet, CANopen Extender, NEMA box | WRC-CANX-NEM-DN |
| SDS Extender, NEMA box | WRC-CANX-NEM-SD |
| DeviceNet, CANopen Extender, Fiber Optic, NEMA box | WRC-CANR-DF-DN |