

## L5351 DeviceNet LinkCard

### AUDIENCE

This document addresses the use of the SSD Link L5351 DeviceNet LinkCard with masters such as PLC DeviceNet Scanners. As its scope is limited to the unique aspects of such applications, it assumes the reader has a working knowledge of the individual components which will not be re-documented here. In particular, the reader should be familiar with:

SSD Link (function blocks, connections, etc.) and the use of its configuration tool ConfigEd;

General DeviceNet concepts and terminology;

Configuration and use of the chosen DeviceNet master.

### GENERAL DESCRIPTION

The L5351 DeviceNet LinkCard is a flexible gateway between DeviceNet and its host Link2 module and fiber-optic ring. It operates as a “Group 2 Only Slave” on DeviceNet and supports both “Explicit” and “Polled I/O” messaging formats using any of the three DeviceNet speeds (125K, 256K and 512K baud) and 64 addresses (MacID’s).

Because some configuration is required in the host Link2 module to route data to and from Link function blocks, the L5351 is configured using Link’s ConfigEd software package. The L5351 is not configurable via the DeviceNet Manager software.

### CONFIGURATION BASICS

Configuring the L5351 consists of two basic parts. The first is a configuration for the host Link module which provides for the definition of “register” function blocks and their and interconnection with other Link blocks for signal-processing or transmission over the Link fiber-optic ring. (The L5351 appears essentially as an I/O device to the rest of the Link system.) The second part of an L5351 configuration resides in the DeviceNet master (such as a PLC DeviceNet Scanner module) and defines the number and type of parameters to transfer and how to map them to PLC memory.

Obviously, these two parts are related. The parameters defined by the Link configuration must match what the scanner is told to expect in number, type, and order. Mismatches can prevent data transfers in either or both directions and result in master error messages complaining of “incorrect data length”.

Because of this interrelation, it is advised that the reader plan the data exchange process and consider the requirements of both parts at the outset.

### DATA REPRESENTATION

The primary consideration when designing-in an L5351 is the data representation. Three data types are supported: Bit, Unsigned Integer, and Signed Integer. In Link, these map to Logics, Values, and Ordinals and each type is created from the “Block->LinkCard/DNet...” menu. On DeviceNet, these types map to Booleans (0 - 1), unsigned 16-bit integers (0 - \$FFFF), and signed 16-bit integers (\$8001 - \$0000 - \$7FFF). (The

code \$8000 is not returned from Signed integer parameters because Link omits it to make the positive and negative ends symmetrical across zero. The L5351 returns \$8001 for negative full scale and clamps any \$8000 consumed from DeviceNet to \$8001 before delivering it to the Link host. Unsigned integers are transferred unchanged.)

Each DeviceNet Link function block holds a group of 16 parameters of the selected type. Each parameter has both an input and an output connector. As with other “gateway” style Link blocks, the inputs and outputs are independent. Data sent to a parameter input (on the left side of the function block) are “produced” onto the DeviceNet bus upon request by the master. Data “consumed” off the DeviceNet bus trigger function block outputs (on the right side of the function block) in Link. The parameters do not “pass through” the Link block.

Data on DeviceNet are “packed”, meaning that each signed and unsigned register block transfers 32 bytes of data while each bit register block transfers 2 bytes. It is important to understand this point in order to make the most efficient use of the bus bandwidth.

## LINK CONFIGURATION

The engineer creates Link DeviceNet register function blocks of the type and number needed to handle the parameters bearing in mind that unused parameters waste space in the I/O messages on the bus. The best performance will only be achieved when registers are fully used.

Each register block must be configured with the desired DeviceNet “instance” number (1-255) which must be unique within its type. Instance numbers are used directly only in Explicit messages. Polled I/O does not directly refer to instance numbers. (See “Explicit DeviceNet Messages” below.)

Finally, the engineer creates a single “LinkCard/L5351 DeviceNet” function block, defines the card’s site, Mac ID and DeviceNet baud rate, and then adds each register block (described above) to the Register List. Each active register block must appear exactly once in this list. (Any register function blocks that are left out of this list will essentially not exist as far as DeviceNet is concerned.)

The L5351, on startup, creates an “I/O Assembly” (Class 4, Instance 1) to represent the packed contents of each register block in the order defined in the Register List in the Link configuration. The default Polled I/O connection points to the data attribute (#3) of this instance.

## DEVICENET CONFIGURATION

Using the DeviceNet Manager the engineer configures the PLC scanner to “allocate” the target L5351’s “Polled I/O Connection Set”. (Relevant MacIDs, the bus baud rate, and the L5351’s “Group 2 Only Slave” status must also be setup.) Then, the scanner is configured to map a region of PLC memory to this polled I/O data. Here, the total byte-length and parameter order of the I/O Assembly must be known in order to map them to PLC memory in a meaningful way.

Once configured and placed in Run mode (with the L5351 also configured and running), the Polled I/O process should begin and bridge the mapped PLC memory to the Link parameters. Because the scanner represents the targeted data as PLC memory, its operation and that of the PLC itself are functionally separated in terms of update rates and processor loads.

## EXPLICIT DEVICENET MESSAGES

If the DeviceNet master supports Explicit messaging, then individual parameters can be accessed via the Get and Set Attribute Single services. The explicit message must include the register’s class number (Boolean [100], unsigned 16-bit integer [101], and signed 16-bit integer [102]), instance number (as configured in the register’s Link function block), and attribute number (1-16, specifying a parameter in the register).

As with Polled I/O, Set Attribute Single operations must convey the correct amount of data. Both the signed and unsigned types use 2 bytes per parameter. The Bit type uses the least significant bit of a single data byte (all other bits are ignored during a Set operation and returned as zero during Get operations). The L5351 will reject Sets containing an

incorrect number of bytes. Similarly, if the master is misconfigured regarding a parameter's byte-size, it may reject the L5351's responses to Get operations.

Although Explicit messaging is fully supported, the explicit addressing information (from which it gets its name) is overhead present in each message which makes it a less efficient protocol than Polled I/O. (The explicit protocol is also acknowledged in a more verbose way.) Parameters that are read or written on a continuous basis should be handled using Polled I/O for maximum performance.

### **PERFORMANCE ESTIMATES**

To optimize the load on the Link host, outputs from Link register function blocks only trigger on-change. This allows for fast I/O on the DeviceNet bus without loading the host Link module with unchanged data.

As the L5351's processing and data are handled using its own CPU and memory, the Link function blocks are reasonably lightweight. A L5391 configuration memory will hold more than 40 register blocks of any one type with each block having 16 connections. This translates to more than 600 word-wide parameters. (More connections per register or the use of multiple register types will reduce this number somewhat. Use of an L5300 host instead of the L539x will increase it.)

Beyond memory, the Link system is generally quoted as processing 1 average Link connection per ms which yields a throughput of roughly 1000 messages per second at the Link function block interface. At the DeviceNet bus, again due to the its dedicated CPU, the L5351 is capable of using the full bus bandwidth at all baud rates.

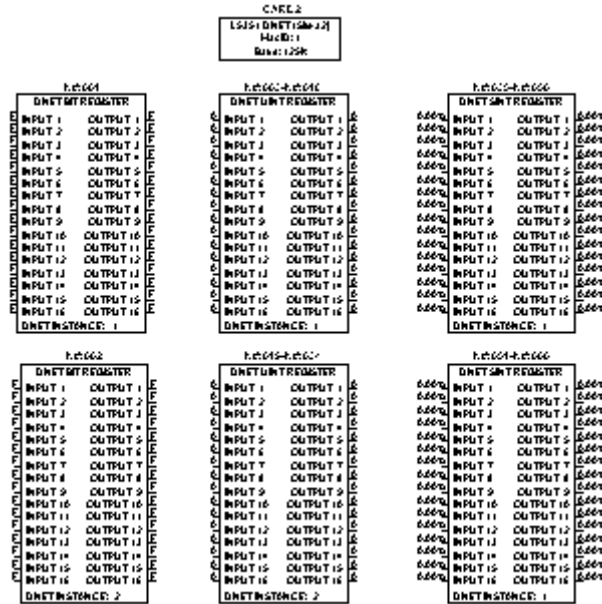
## **Overview**

In the following example, a I2 module with a L5351 DeviceNet LinkCard installed will be configured to communicate to a PLC-5 using a polled I/O DeviceNet connection. The transfer will require 132 bytes of data to be transferred in both directions from a 1771-SDN DeviceNet scanner module to the LinkCard.

### ***DeviceNet™* LINKConfiguration**

The following is a I2 DeviceNet configuration which transfers:

- 32 bit inputs and 32 bit outputs
- 32 unsigned integer inputs and 32 unsigned outputs
- 32 signed integers and 32 unsigned integers



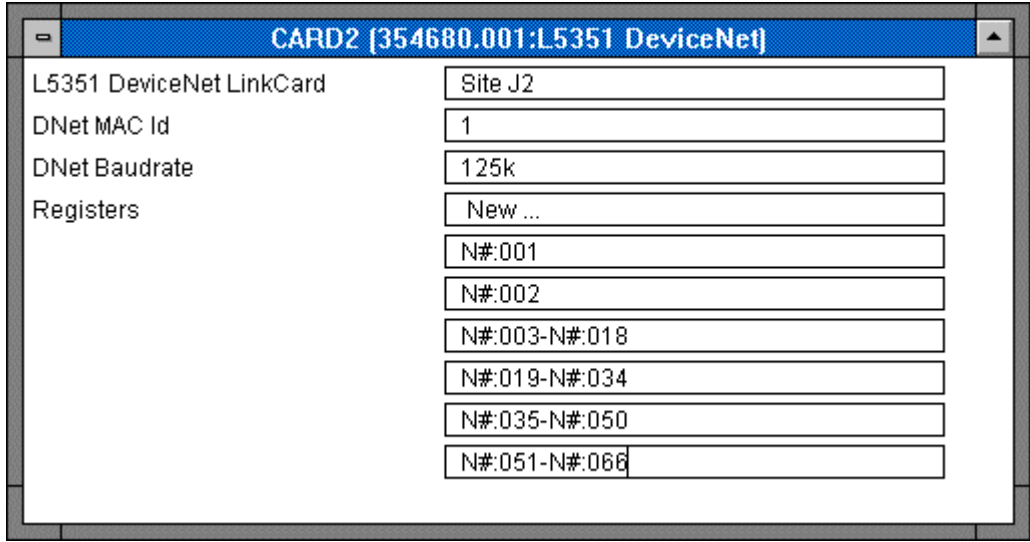
### DeviceNet™ Slot Controller

The slot controller function block is where:

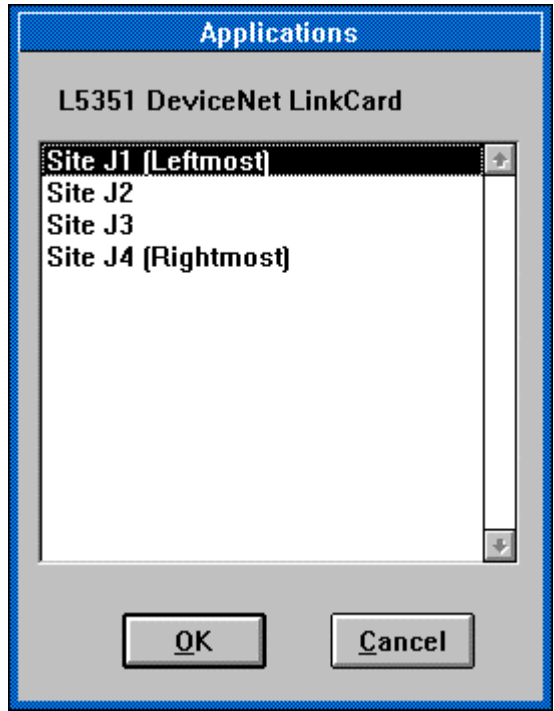
- select the Ik slot position
- define the DeviceNet address (MAC ID in DeviceNet terms)
- configure the baud rate for the DeviceNet Ik
- establish the mapping of the data transfers from DeviceNet



### Slot Controller Setup Window

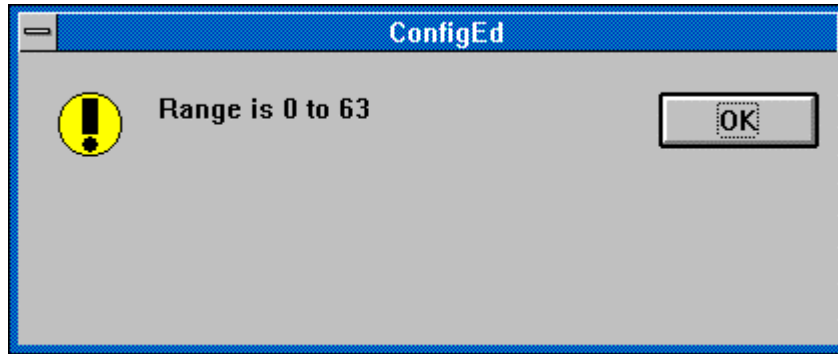


**LINKCard Slot Location**



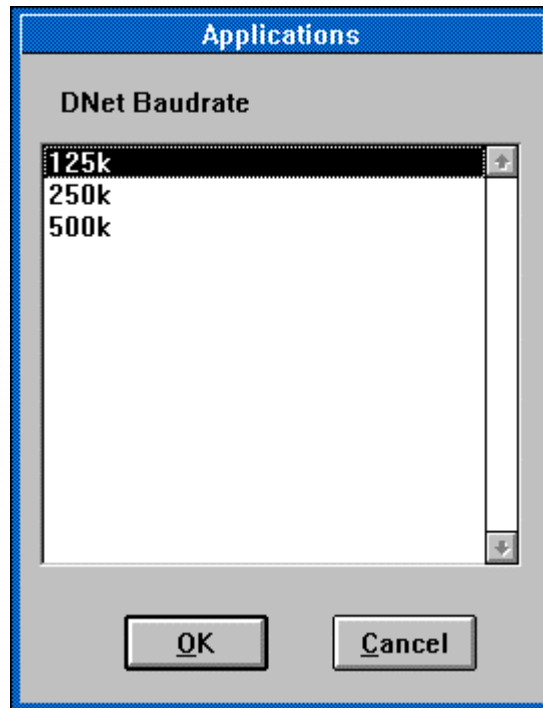
**DeviceNet™ MAC Id Out-of-Range Dialog**

DeviceNet is limited to 64 nodes per network resulting in valid MAC ID's of 0 through 63. If you try to enter an invalid Mac ID, the following dialog box will appear:



### *DeviceNet™* Baudrate Setting Window

DeviceNet supports three different baud rates; however, transmission distance varies inversely with the baud rate selected

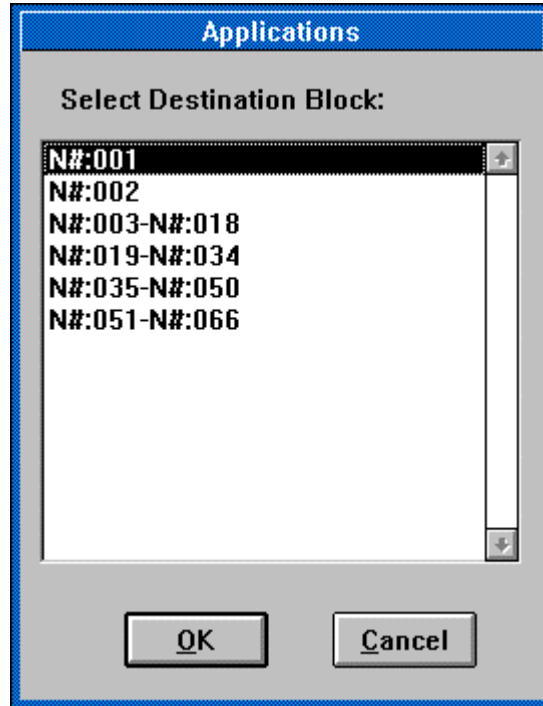


The distances quoted by the ODVA specification are listed below:

DeviceNet Baud Rate	Thick Trunk	Thin Trunk	Maximum Drop	Cumulative Drop
125 Kbaud	500m (1640 ft)	100m (328 ft)	6 m (20 ft)	156 m (512 ft)
250 Kbaud	250m (820 ft)	100m (328 ft)	6 m (20 ft)	78 m (256 ft)
500 Kbaud	100 m (328 ft)	100m (328 ft)	6 m (20 ft)	39 m (128 ft)

### Register Insertion Selection Box

The order in which DeviceNet function blocks are placed in the Register List determines their location in the exchange between the DeviceNet scanner and the LinkCard. In this example, the DeviceNet function blocks were renamed (Ctrl+L) to indicate their mapping to the PLC memory. This convention makes insertion into this block more intuitive and maintains the self-documenting feature of ConfigEd.



### Bit Register Function Block

The Bit Register Function Block has 16 boolean (true/false) inputs/outputs. The outputs are presetable.

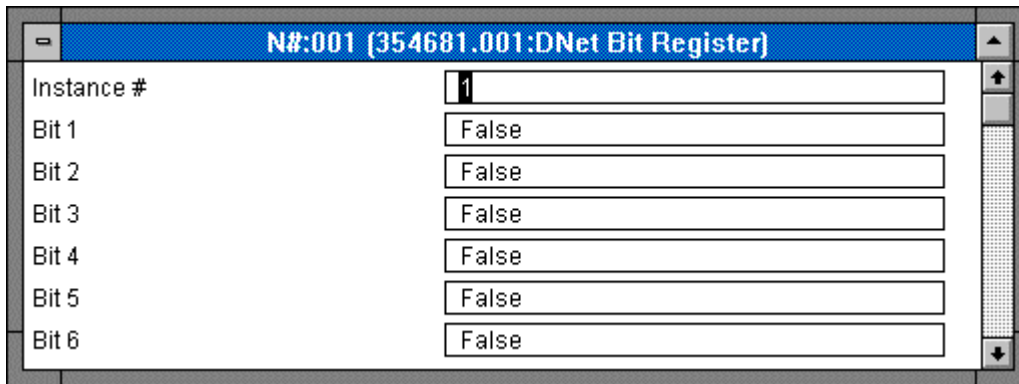


**N#:001**

DNET BIT REGISTER	
INPUT 1	OUTPUT 1
INPUT 2	OUTPUT 2
INPUT 3	OUTPUT 3
INPUT 4	OUTPUT 4
INPUT 5	OUTPUT 5
INPUT 6	OUTPUT 6
INPUT 7	OUTPUT 7
INPUT 8	OUTPUT 8
INPUT 9	OUTPUT 9
INPUT 10	OUTPUT 10
INPUT 11	OUTPUT 11
INPUT 12	OUTPUT 12
INPUT 13	OUTPUT 13
INPUT 14	OUTPUT 14
INPUT 15	OUTPUT 15
INPUT 16	OUTPUT 16
DNET INSTANCE: 1	

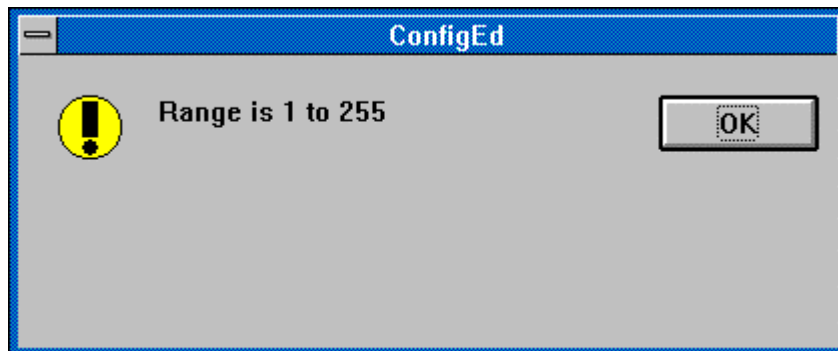
### Bit Register Configuration

The only configuration required is the setting of the DeviceNet instance to support explicit addressing.



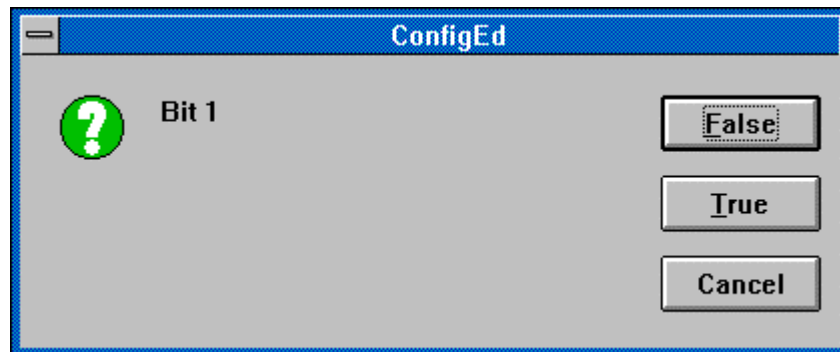
### Instance Error Dialog Window

Valid ranges for instances per the ODVA specification are 1 to 255.



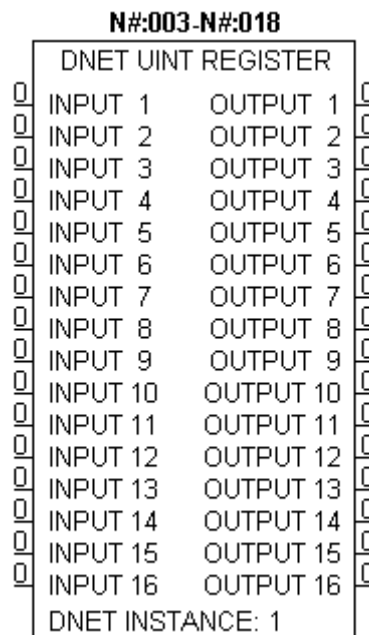
## Bit Preset Dialog Window

As mentioned before, the outputs 1 through 16 may be preset *True* or *False*.



## Unsigned Integer Register Function Block

The Unsigned Integer Register Function Block has 16 unsigned integer inputs/outputs (0 through 65535). Again, the outputs are presettable.



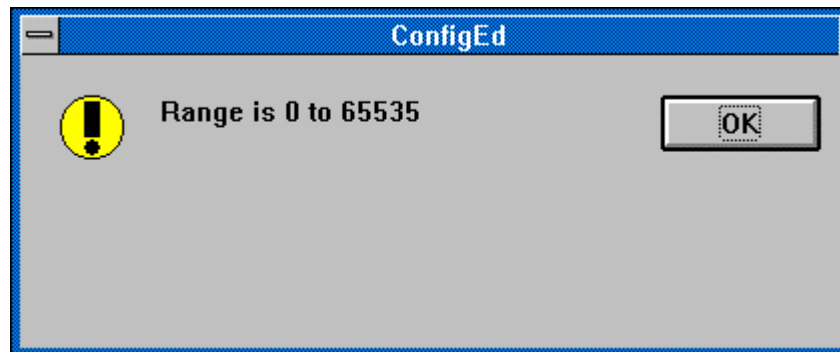
## Unsigned Integer Register Configuration

Again, the only configuration required is the setting of the DeviceNet instance to support explicit addressing.

N#:003-N#:018 [354682.001:DNet UInt Register]	
Instance #	<input type="text" value="1"/>
Integer Data 1	<input type="text" value="0"/>
Integer Data 2	<input type="text" value="0"/>
Integer Data 3	<input type="text" value="0"/>
Integer Data 4	<input type="text" value="0"/>
Integer Data 5	<input type="text" value="0"/>
Integer Data 6	<input type="text" value="0"/>

### Unsigned Integer Preset Out-of-Range Dialog

If a value less than 0 or greater than 65535 is entered as a preset, ConfigEd will report an error as follows:



### Signed Integer Register Function Block

The Signed Integer Register Function Block has 16 signed integer inputs/outputs (-100% through +100%). Again, the outputs are presettable.

**N#:035-N#:050**

DNET SINT REGISTER			
0.00%	INPUT 1	OUTPUT 1	0.00%
0.00%	INPUT 2	OUTPUT 2	0.00%
0.00%	INPUT 3	OUTPUT 3	0.00%
0.00%	INPUT 4	OUTPUT 4	0.00%
0.00%	INPUT 5	OUTPUT 5	0.00%
0.00%	INPUT 6	OUTPUT 6	0.00%
0.00%	INPUT 7	OUTPUT 7	0.00%
0.00%	INPUT 8	OUTPUT 8	0.00%
0.00%	INPUT 9	OUTPUT 9	0.00%
0.00%	INPUT 10	OUTPUT 10	0.00%
0.00%	INPUT 11	OUTPUT 11	0.00%
0.00%	INPUT 12	OUTPUT 12	0.00%
0.00%	INPUT 13	OUTPUT 13	0.00%
0.00%	INPUT 14	OUTPUT 14	0.00%
0.00%	INPUT 15	OUTPUT 15	0.00%
0.00%	INPUT 16	OUTPUT 16	0.00%
DNET INSTANCE: 1			

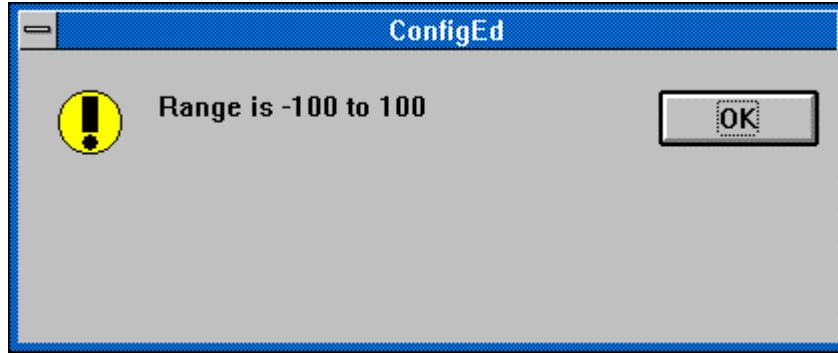
### Signed Integer Register

Again, the only configuration required is the setting of the DeviceNet instance to support explicit addressing.

N#:035-N#:050 [354683.001:DNet SInt Register]	
Instance #	1
Value Data 1 (%)	0
Value Data 2 (%)	0
Value Data 3 (%)	0
Value Data 4 (%)	0
Value Data 5 (%)	0
Value Data 6 (%)	0

### Signed Integer Preset Out-of-Range Dialog

If a value less than -100% or greater than 100% is entered as a preset, ConfigEd will report an error as follows:



## DeviceNet™ Scanner Configuration Project Window

Project Name: **MASTERS**

**Network:** 1771\_SDN      **Network Description:** L5391 to 1771-SDN Scanner

Net Data Rate: 125 k      **Add Network...**

Number of Devices: 3

**Add Device...**    **Config Device...**      **Online Build**    **Stop Build**

Node	Vendor	Product Name	Node Name
00	Allen-Bradley Company	1771-SDN Scanner Module	PLC5
01	Eurotherm Controls Lim	DeviceNet LINKCard	L5351
02	Allen-Bradley Company	1770-KFD RS232 Interface	Computer

Catalog Number: 1771-SDN      Device Type: Communication Adapter  
 Major Revision: 3  
 Node Description: Scanner Module for PLC-5.

## Config Device... Window

**Module Settings**

Project Name: PLC5TEST  
 Module Name: PLC5

Access: DeviceNet

Density: 1 Slot : Double Density

Rack: 0      Group: 2

Slot: 0

**PLC Interface Addresses**

	Input	Output
BXfer 62	N9:0	N10:0
BXfer 61	N9:62	N10:62
BXfer 60	N9:123	N10:123
BXfer 59	N9:183	N10:183
BXfer 58	N9:242	N10:242
BXfer 57	N9:300	N10:300

Channel Select: A      Network Name: 2@125K  
 Channel Node Address: 0

**Channel A Settings**

I/O Comms:  Enabled  
 Interscan Delay: 2 ms.  
 Bkgd Poll Ratio: 1

**Load From**

SDN    File...

**Save To**

SDN    File...

**Select Defaults**

Module    Chan

**Close**    **Help**      **Assign Names from Project**    **Edit Scan List...**

## Edit Scan List... Window

Node	Name	Mapped	Active	Rx Size	Tx Size	Type
A01	L5351	Yes/Yes	Yes	132	132	P

**Edit Selection**

Prod Type: \_\_\_\_\_

Vendor: \_\_\_\_\_

Cat No: \_\_\_\_\_

Revision: \_\_\_\_\_

Active In Scanlist

**Electronic Key**

Device Type

Vendor

Product No.

**Load From**

**Save To**

**Add Devices From**

**Scan List Tools**

*Datatable map... Window*

**Inputs from LINK addresses N9:001-N9:061**

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
N9:000	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
N9:001	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:002	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:003	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:004	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:005	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:006	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:007	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:008	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:009	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1

Display Mode:  Data Entry  Browse Double-click on word to display bit mapping

Device Select: A01 L5351 Generic L5351

Data Map:  Input  Output

Map Segment:  1  2  3  4

Map Data From: Poll Message Byte 0 Bit 0

Map Data To: Block Transfer 62 N9: 1 Bit 0 No. Bits 976

**Inputs from LINK addresses N9:062-N9:066**

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
N9:062	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:063	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:064	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:065	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:066	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N9:067	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N9:068	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N9:069	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N9:070	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N9:071	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Display Mode:  Data Entry  Browse Double-click on word to display bit mapping

Device Select: A01 L5351 Generic L5351

Data Map:  Input  Output Apply Segment

Map Segment:  1  2  3  4 Delete Segment

Map Data From: Poll Message Byte 122 Bit 0

Map Data To: Block Transfer 61 N9: 62 Bit 0 No. Bits 80

Close Help Print to File

### Inputs to LINK Addresses N10:001-N10:061

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
N10:000	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
N10:001	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:002	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:003	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:004	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:005	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:006	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:007	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:008	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:009	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1

Display Mode:  Data Entry  Browse Double-click on word to display bit mapping

Device Select: A01 L5351 Generic L5351

Data Map:  Input  Output Apply Segment

Map Segment:  1  2  3  4 Delete Segment

Map Data To: Poll Message Byte 0 Bit 0

Map Data From: Block Transfer 62 N10: 1 Bit 0 No. Bits 976

Close Help Print to File

### Inputs to LINK Addresses N10:062-N10:066



	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
N10:062	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:063	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:064	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:065	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:066	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1
N10:067	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N10:068	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N10:069	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N10:070	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N10:071	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.

Display Mode:  Data Entry  Browse Double-click on word to display bit mapping

Device Select:

Data Map:  Input  Output Apply Segment

Map Segment:  1  2  3  4 Delete Segment

Map Data To:  Byte  Bit

Map Data From:  N10:  Bit  No. Bits

Close Help Print to File

## PLC Programming

Rung 2:0

Enable	Reads	Writes	Reads	Writes	Read	
Block 62	N9:000	N10:000	N9:062	N9:062	Transfer	
Reads	Thru	Thru	Thru	Thru	N9:000	
SW 00/10	N9:061	N10:061	N9:122	N9:122	Thru	
	Enabled	Enabled	Enabled	Enabled	N9:061	
I:001	N11:0	N11:5	N11:10	N11:15	+BTR-----+	
+----] [-----]/[-----]/[-----]/[-----]/[-----]					++BLOCK TRANSFER READ	+- (EN)++
00	15	15	15	15	Rack	00
					Group	2+- (DN)
					Module	0
					Control block	N11:0+- (ER)
					Data file	N9:0
					Length	62
					Continuous	N
					-----+	
					Read	
					Block 62	
					Enabled	
					LED 00/10	
					O:001	
					----- ( )-----	
						00

Rung 2:1

Enable	Reads	Writes	Reads	Writes	Write	
Block 62	N9:000	N10:000	N9:062	N9:062	Transfer	
Writes	Thru	Thru	Thru	Thru	N10:000	
SW 00/10	N9:061	N10:061	N9:122	N9:122	Thru	
	Enabled	Enabled	Enabled	Enabled	N10:061	
I:001	N11:0	N11:5	N11:10	N11:15	+BTW-----+	
+----] [-----]/[-----]/[-----]/[-----]/[-----]					++BLOCK TRANSFER WRITE	+- (EN)++
01	15	15	15	15	Rack	00
Disable					Group	2+- (DN)
Channel 1					Module	0
SW 04/14					Control block	N11:5+- (ER)
					Data file	N10:0
					Length	62
I:001					Continuous	N
					-----+	
04					Write	
					Block 62	
					Enabled	
					LED 01/11	
					O:001	
					----- ( )-----	
						01

Rung 2:2

Enable	Reads	Writes	Reads	Writes	Read
Block 62	N9:000	N10:000	N9:062	N9:062	Transfer
Reads	Thru	Thru	Thru	Thru	N9:062
SW 02/12	N9:061	N10:061	N9:122	N9:122	Thru
	Enabled	Enabled	Enabled	Enabled	N9:122
I:001	N11:0	N11:5	N11:10	N11:15	+BTR-----+
+----] [-----]/[-----]/[-----]/[-----]/[-----]-----+BLOCK TRANSFER READ +- (EN)++					
02	15	15	15	15	Rack 00
					Group 2+-(DN)
					Module 0
					Control block N11:10+-(ER)
					Data file N9:62
					Length 61
					Continuous N
					+-----+
					Read
					Block 61
					Enabled
					LED 02/12
					O:001
					+------( )-----+
					02

Rung 2:3

Enable	Reads	Writes	Reads	Writes	Write
Block 61	N9:000	N10:000	N9:062	N9:062	Transfer
Writes	Thru	Thru	Thru	Thru	N9:062
SW 03/13	N9:061	N10:061	N9:122	N9:122	Thru
	Enabled	Enabled	Enabled	Enabled	N9:122
I:001	N11:0	N11:5	N11:10	N11:15	+BTW-----+
+----] [-----]/[-----]/[-----]/[-----]/[-----]-----+BLOCK TRANSFER WRITE +- (EN)++					
03	15	15	15	15	Rack 00
					Group 2+-(DN)
					Module 0
					Control block N11:15+-(ER)
					Data file N10:62
					Length 61
					Continuous N
					+-----+
					Write
					Block 61
					Enabled
					LED 03/13
					O:001
					+------( )-----+
					03

```
Rung 2:4
| Disable | Disable |
| Channel 1 | Scanner |
| SW 04/14 | Channel 1 |
| | (Code 90) |
| | |
| I:001 | N10:0 |
+----] [-----+-----+-----( )-----+
| 04 | | 4 | |
| | | Scanner | |
| | | Channel 1 | |
| | | Disabled | |
| | | LED 04/14 | |
| | | | |
| | | O:001 | |
| | | +---( )---+ |
| | | 04 | |
Rung 2:5
| Enable | Scanner |
| Block 62 | Channel 1 |
| Reads | In Run |
| SW 00/10 | Mode |
| | |
| I:001 | N10:0 |
+----] [-----+-----+-----( )-----+
| | 00 | | 0 |
| | Enable | | |
| | Block 62 | | |
| | Writes | | |
| | SW 00/10 | | |
| | | | |
| | I:001 | | |
| | +---] [-----+ | |
| | 01 | | |
| | Enable | | |
| | Block 62 | | |
| | Reads | | |
| | SW 02/12 | | |
| | | | |
| | I:001 | | |
| | +---] [-----+ | |
| | 02 | | |
| | Enable | | |
| | Block 61 | | |
| | Writes | | |
| | SW 03/13 | | |
| | | | |
| | I:001 | | |
| | +---] [-----+ | |
| | 03 | | |
```

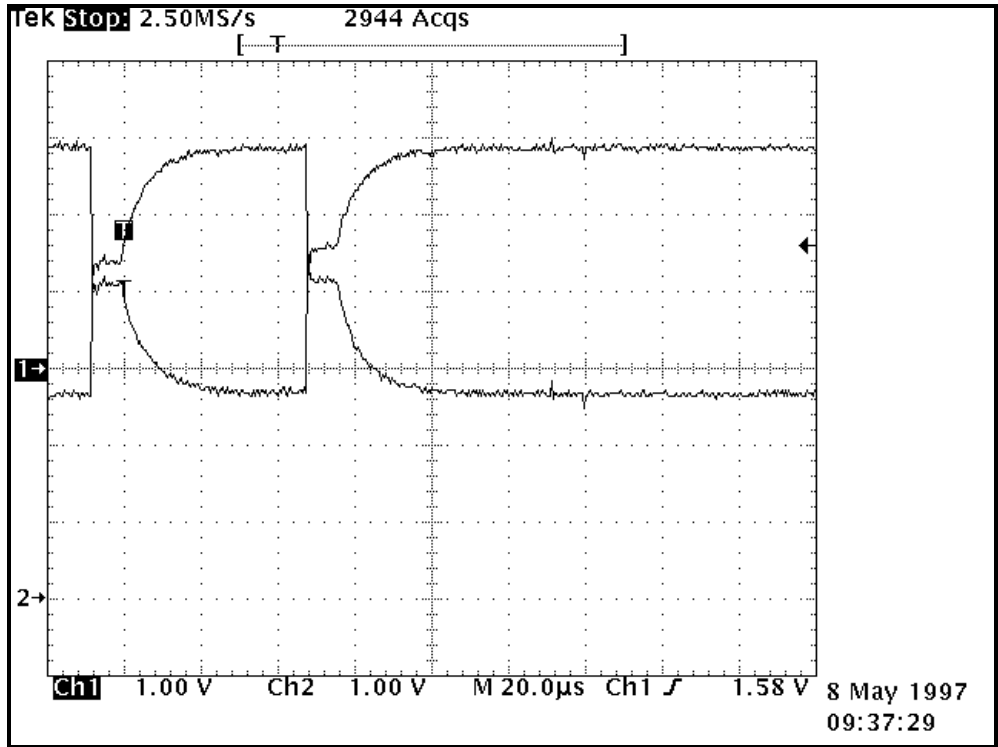
Rung 2:6

	Disable	
	Scanner	
	Channel 2	
	(Code 90)	
	N10:0	
----- ( ) -----		
	5	

Rung 2:7

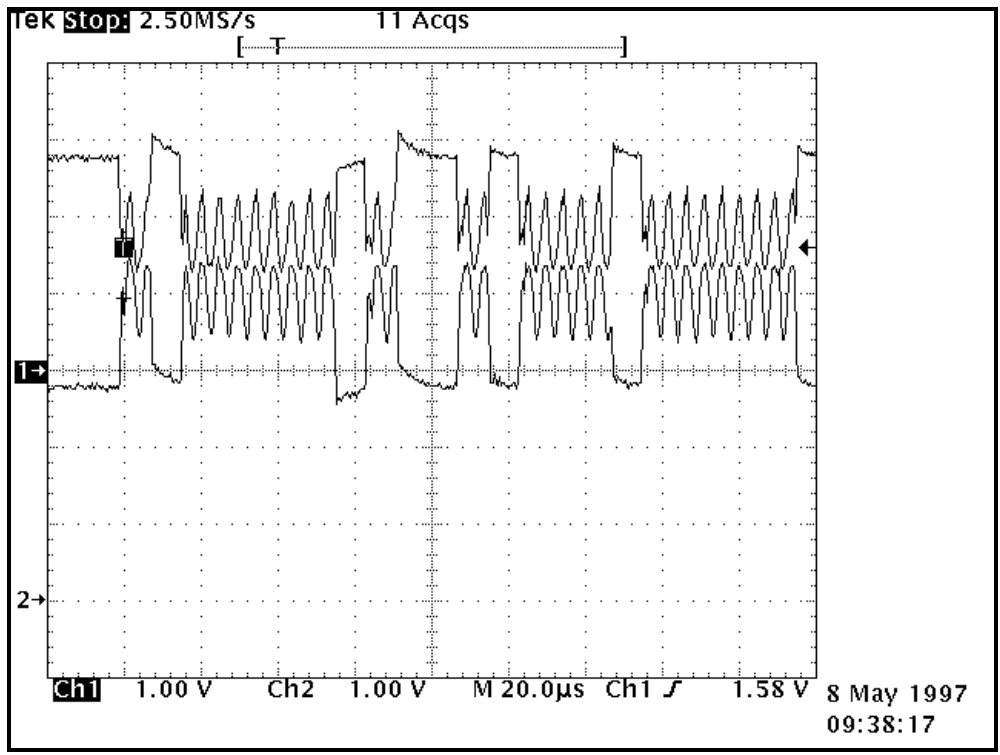
----- [END OF FILE] -----		

**SCOPE TRACES AT DIFFERENT BAUD RATES**  
**Unterminated at both 1771-SDN and L5351 ends @ 125Kbaud**



- Channel 1 = CH (CAN High)**
- Channel 2 = CL (CAN Low)**
- Common = V- (24 VDC Common)**

**Termination resistor (127 Ω) at L5351 end (127 Ω) @ 125Kbaud**

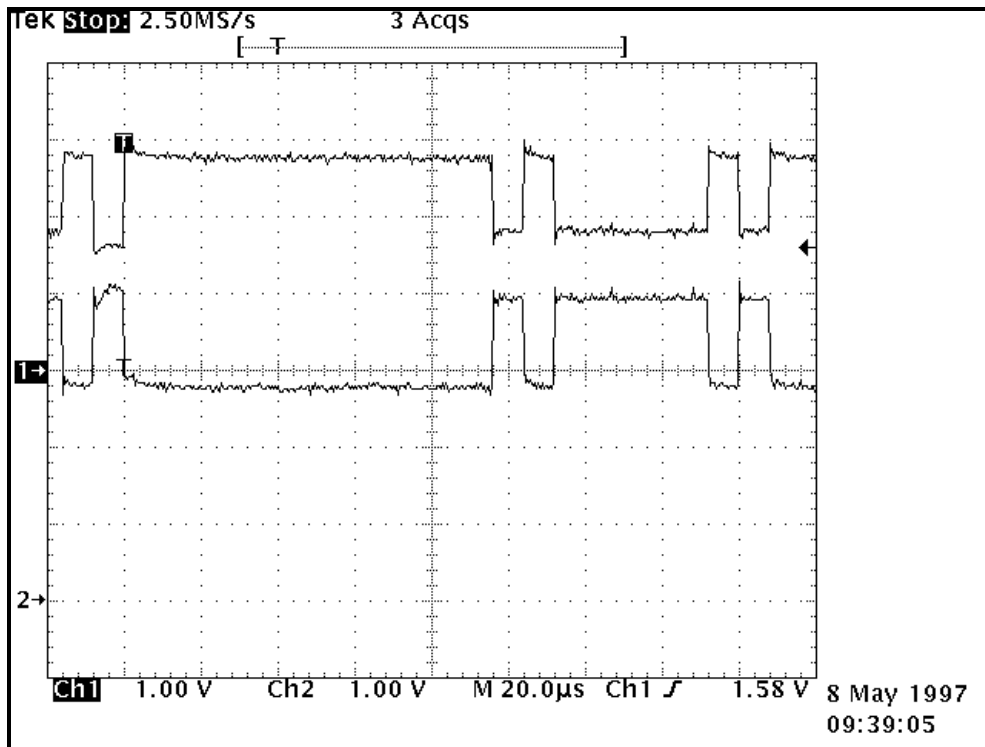


**Channel 1 = CH (CAN High)**

**Channel 2 = CL (CAN Low)**

**Common = V- (24 VDC Common)**

Termination resistors (127 Ω) at both 1771-SDN and L5351 end @ 125Kbaud



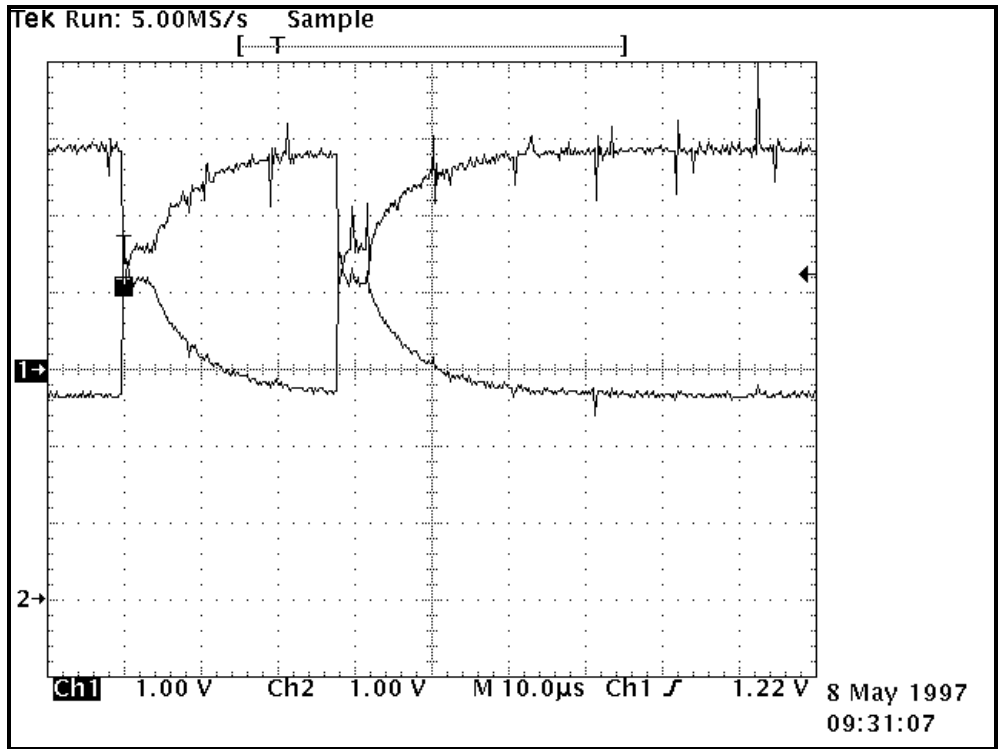
Channel 1 = CH (CAN High)

Channel 2 = CL (CAN Low)

Common = V- (24 VDC Common)



**Unterminated at both 1771-SDN and L5351 end @ 250Kbaud**

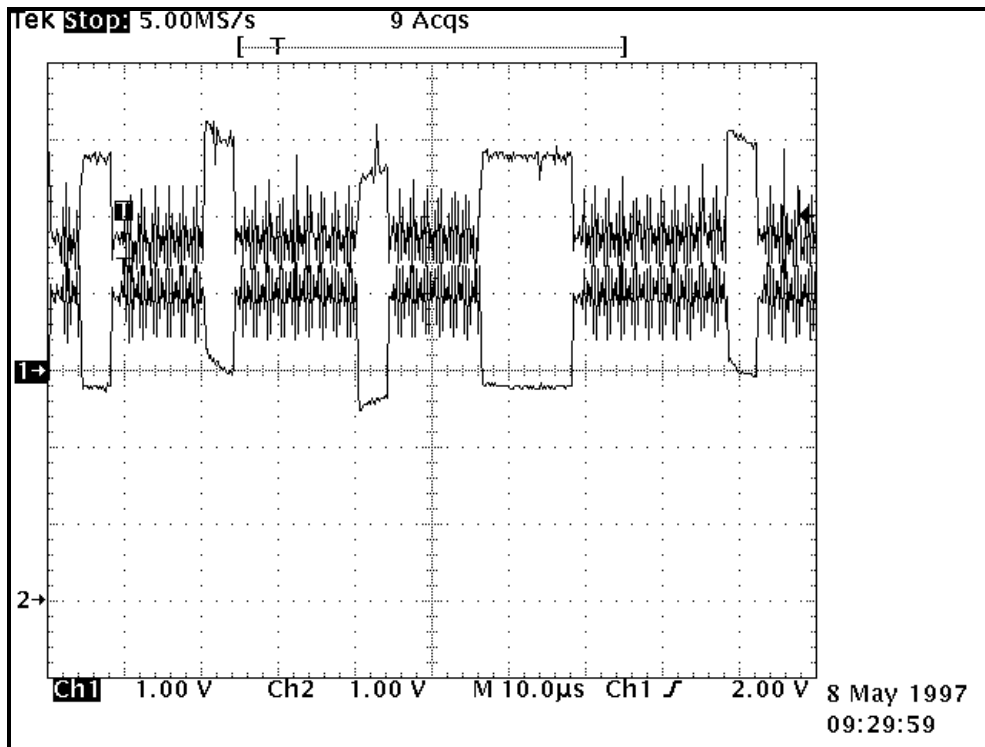


**Channel 1 = CH (CAN High)**

**Channel 2 = CL (CAN Low)**

**Common = V- (24 VDC Common)**

Termination resistor (127 Ω) at L5351 end @ 250Kbaud

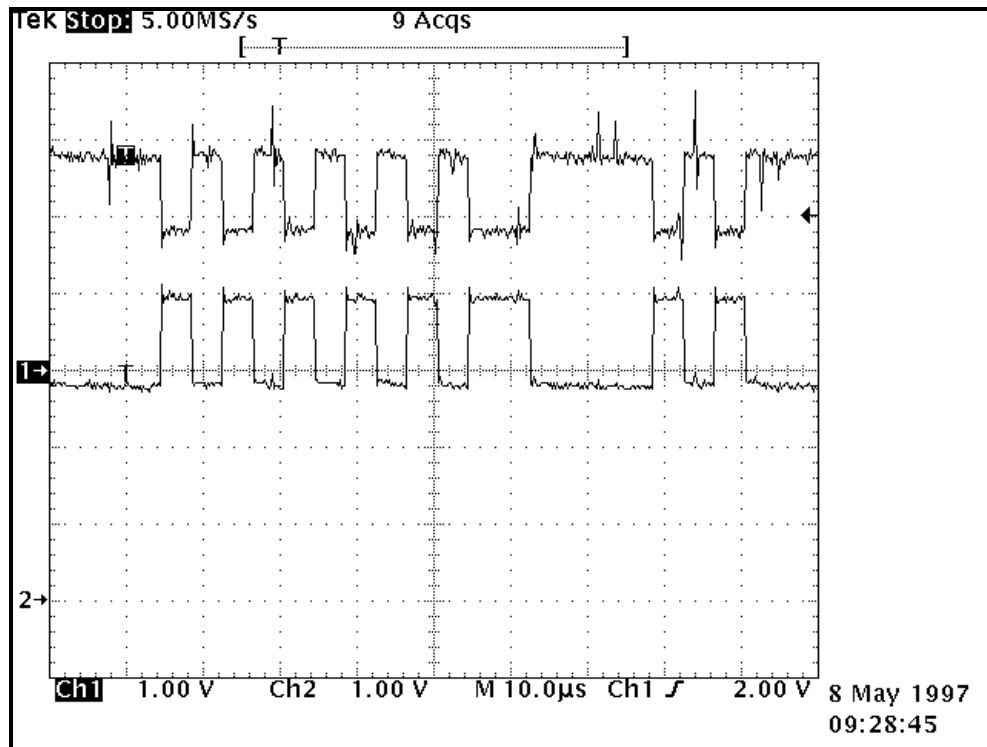


Channel 1 = CH (CAN High)

Channel 2 = CL (CAN Low)

Common = V- (24 VDC Common)

Termination resistors (127 Ω) at both 1771-SDN and L5351 end @ 250Kbaud

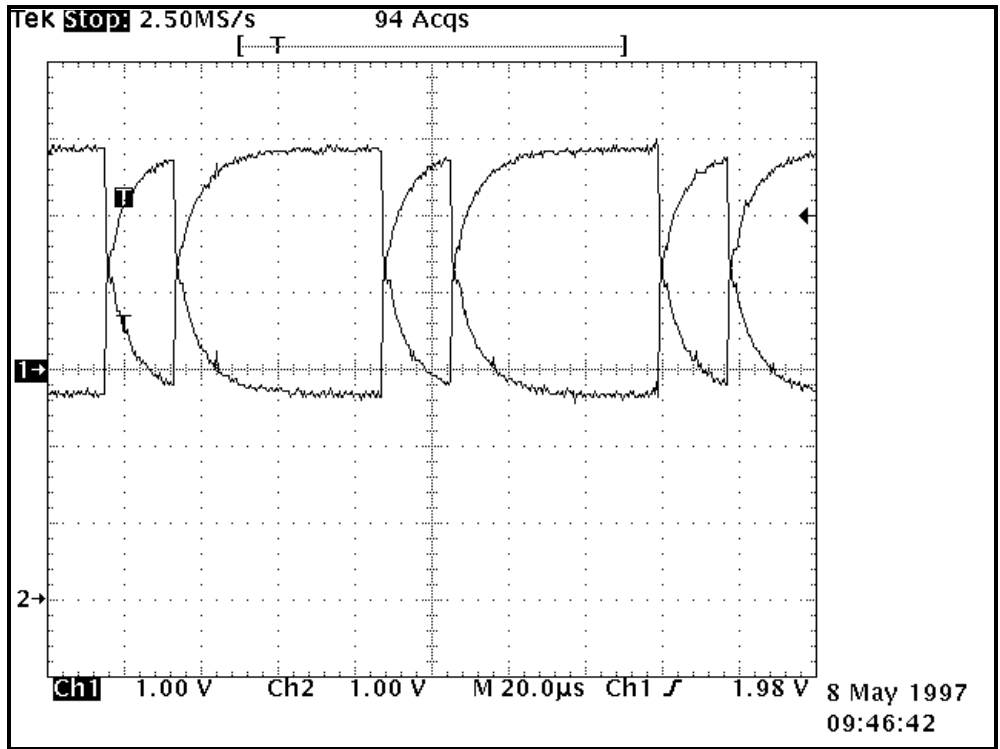


Channel 1 = CH (CAN High)

Channel 2 = CL (CAN Low)

Common = V- (24 VDC Common)

Unterminated at both 1771-SDN and L5351 ends @ 500 Kbaud

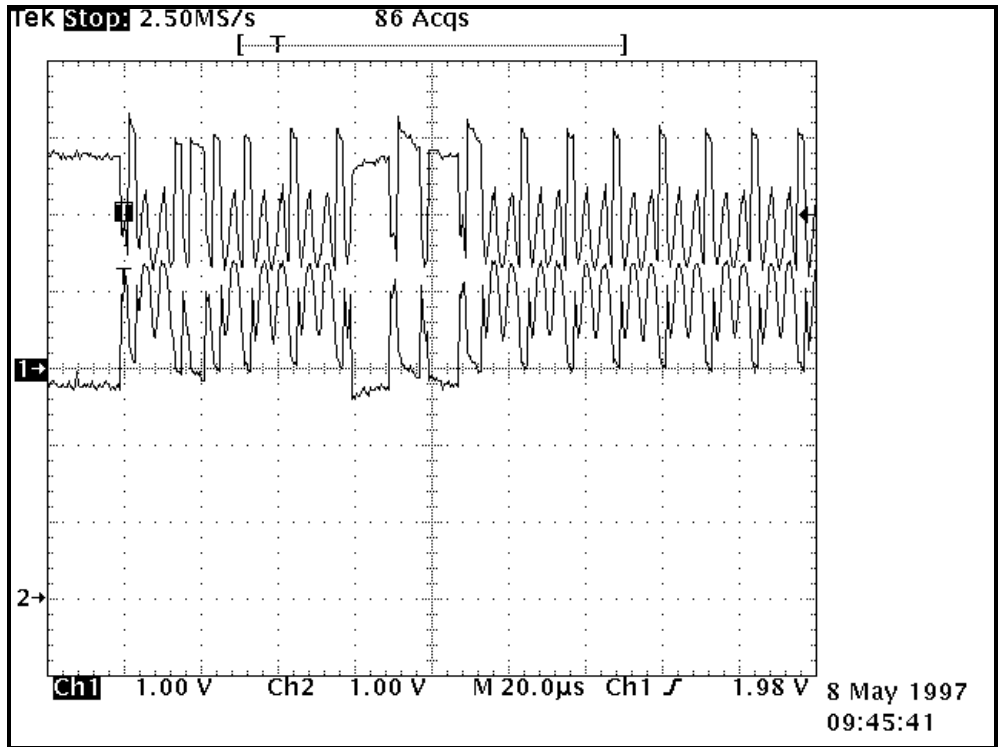


Channel 1 = CH (CAN High)

Channel 2 = CL (CAN Low)

Common = V- (24 VDC Common)

**Termination resistor (127 Ω) at L5351 end (127 Ω) @ 500 Kbaud**

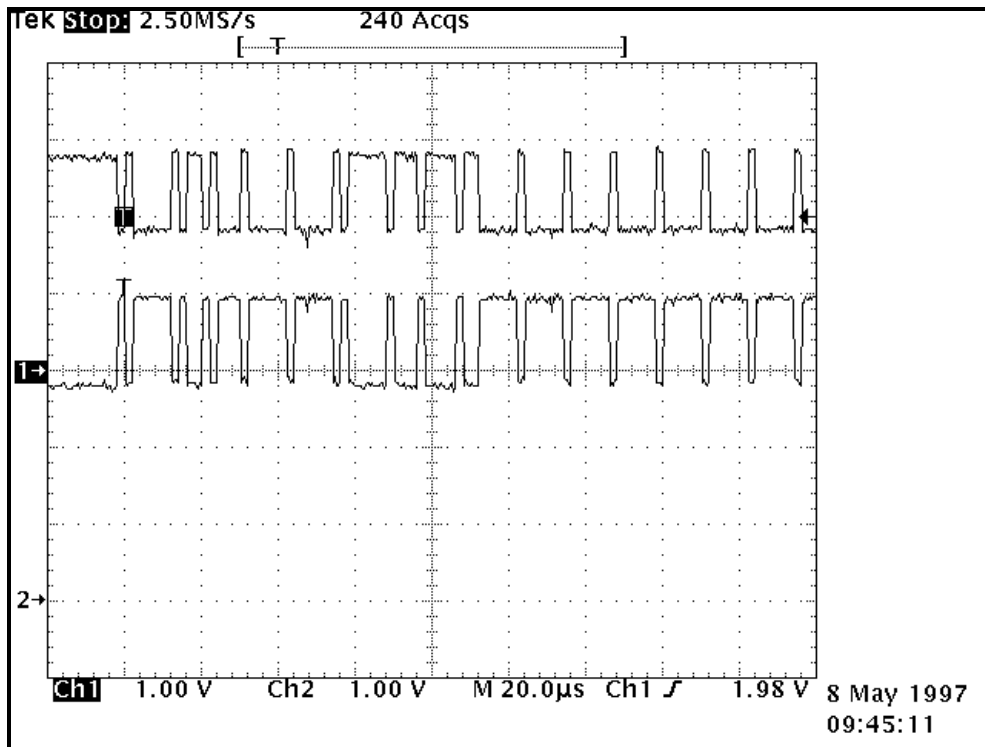


**Channel 1 = CH (CAN High)**

**Channel 2 = CL (CAN Low)**

**Common = V- (24 VDC Common)**

Termination resistors (127 Ω) at both 1771-SDN and L5351 end @ 500 Kbaud



Channel 1 = CH (CAN High)

Channel 2 = CL (CAN Low)

Common = V- (24 VDC Common)