

Configuring TP-78 Devices with LonMaker Browser

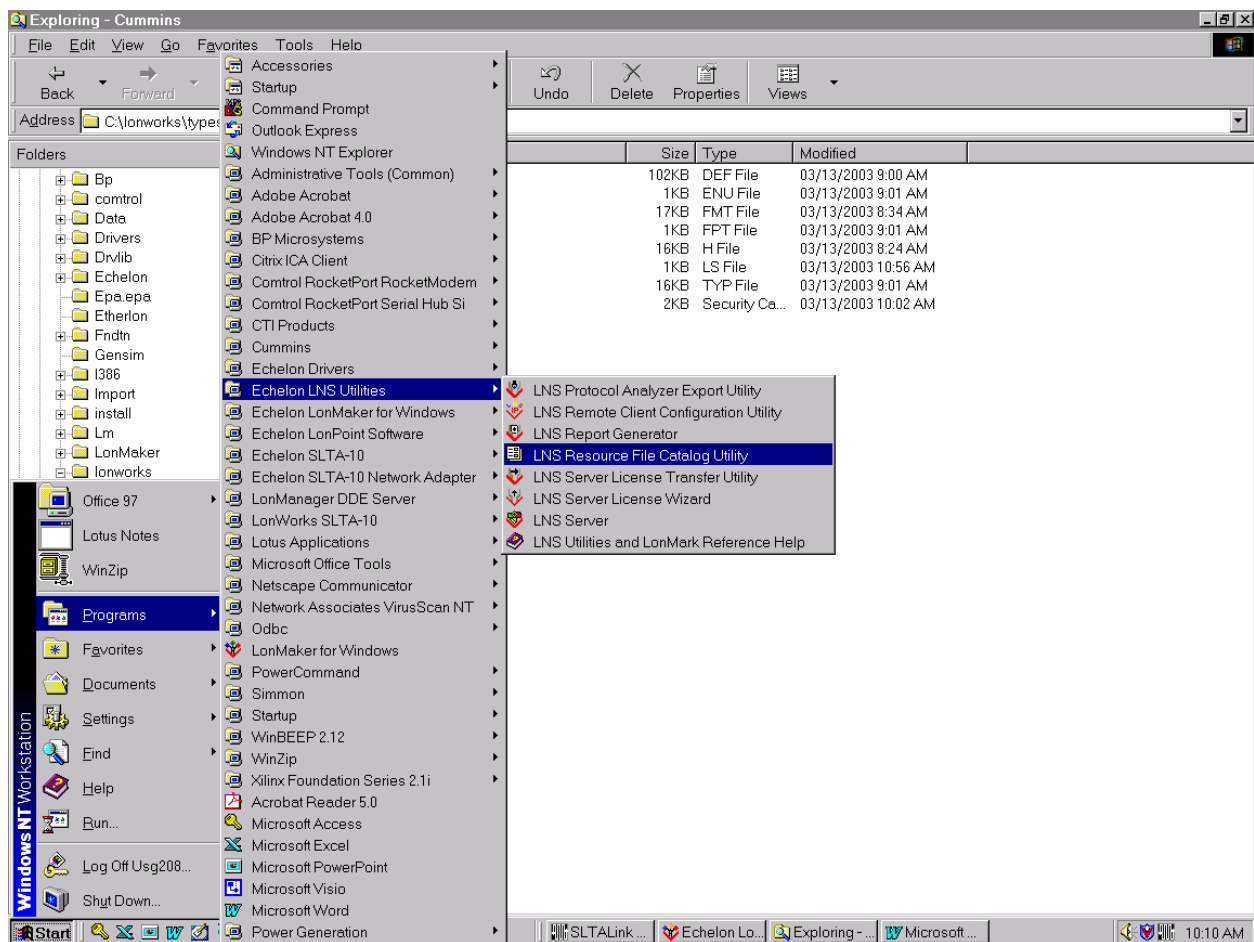
The LonMaker for Windows Browser can write all the information to the devices that PCT can do. By using this method, TP-78 devices can be configured without using PCT or creating the network with LonMaker for DOS. If you are using PCW 1.1 to monitor the network you will still need to use PCT and LonMaker for DOS.

Step 1. Include the Cummins resource files in the LNS resource file catalog.

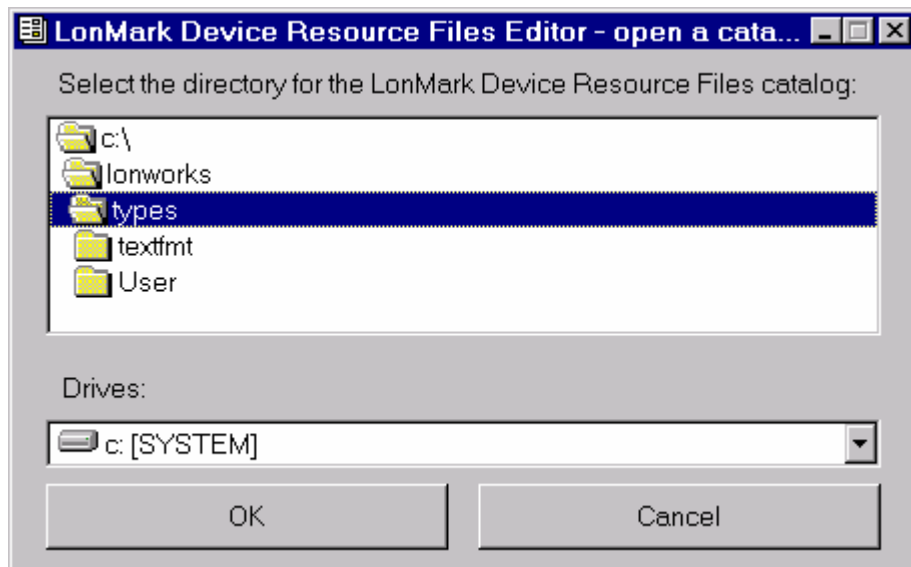
This step is not absolutely necessary but it will make the process easier. The resource files allow you to read and write to the variables in a user defined format, such as integers or character strings. Without using the format files you would be required to write to individual bytes.

Step 1.1 Save the four resource files (Cummins.enu, Cummins.typ, Cummins.fpt, and Cummins.fmt) in your PC. Recommended location is "c:\LonWorks\Types\User\Cummins\".

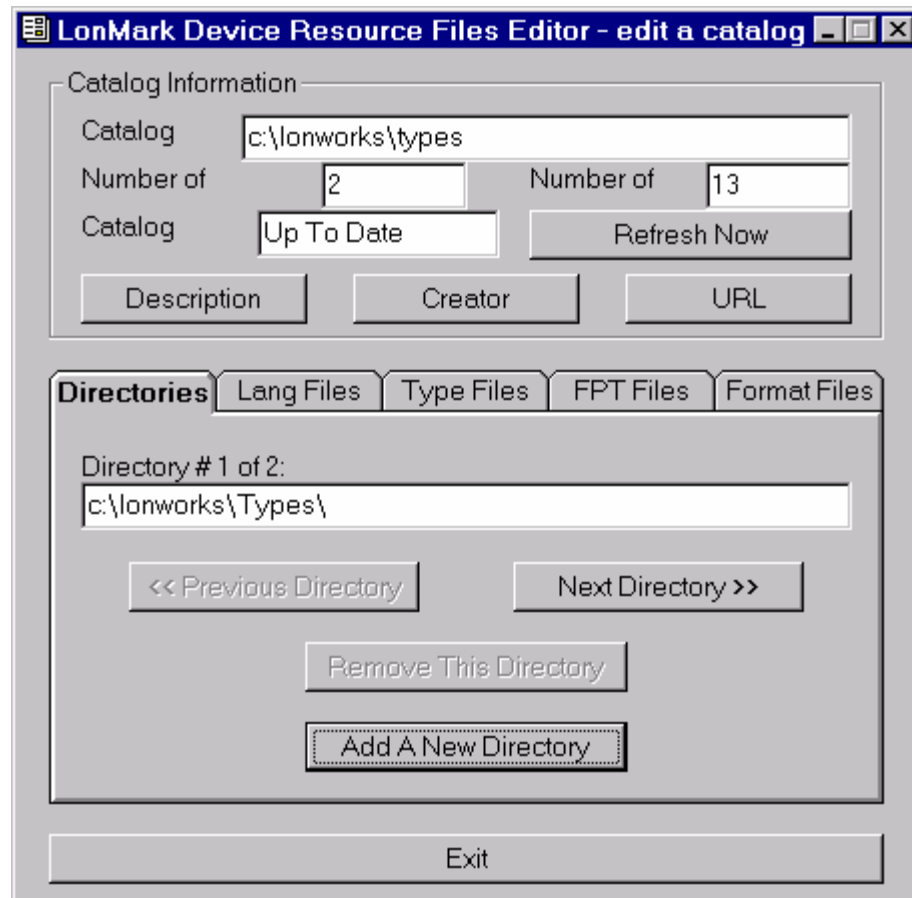
Step 1.2 From the Windows "Start" button, start Echelon's "LNS Resource File Catalog Utility".



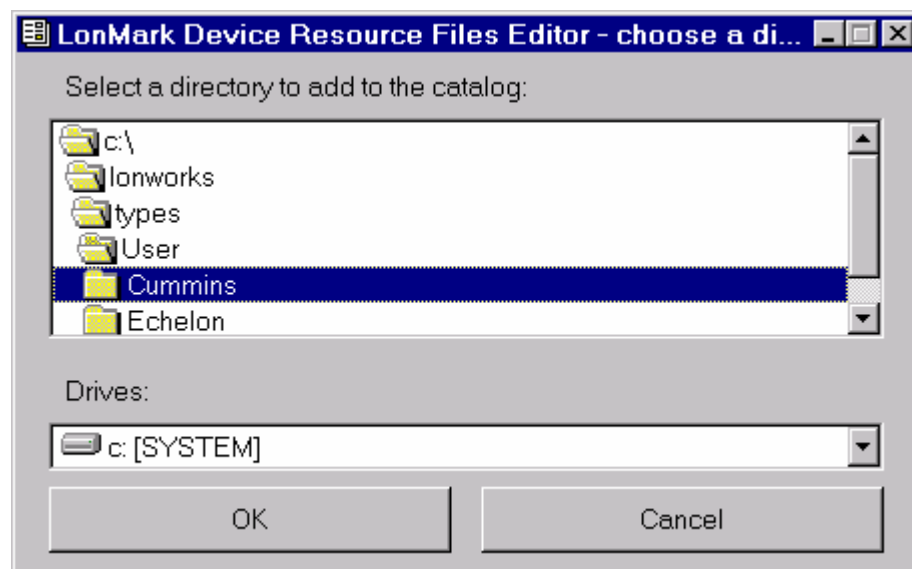
Step 1.3 Navigate to the directory in which the resource files are stored (“c:\Lonworks\Types\” in this example.)
Select “OK”.



Step 1.4 Select “Add A New Directory”.



Step 1.5 Navigate to the directory in which you’ve just created the new resource files. Select “OK”



Step 1.6 Select “Refresh Now”

The screenshot shows a Windows-style application window titled "LonMark Device Resource Files Editor - edit a catalog". The window has a menu bar with a file icon and standard window controls. Below the title bar is a "Catalog Information" section. It contains a "Catalog" text box with the value "c:\lonworks\types", a "Number of" text box with the value "3", and another "Number of" text box with the value "13". Below these is a "Catalog" text box with the text "Needs a Refresh -" in red, and a "Refresh Now" button. At the bottom of this section are three buttons: "Description", "Creator", and "URL". Below the "Catalog Information" section is a tabbed interface with five tabs: "Directories", "Lang Files", "Type Files", "FPT Files", and "Format Files". The "Directories" tab is selected. It shows "Directory # 1 of 3:" followed by a text box containing "c:\lonworks\types\". Below this are four buttons: "<< Previous Directory", "Next Directory >>", "Remove This Directory", and "Add A New Directory". At the very bottom of the window is a large "Exit" button.

LonMark Device Resource Files Editor - edit a catalog

Catalog Information

Catalog c:\lonworks\types

Number of 3 Number of 13

Catalog Needs a Refresh - Refresh Now

Description Creator URL

Directories Lang Files Type Files FPT Files Format Files

Directory # 1 of 3:

c:\lonworks\types\

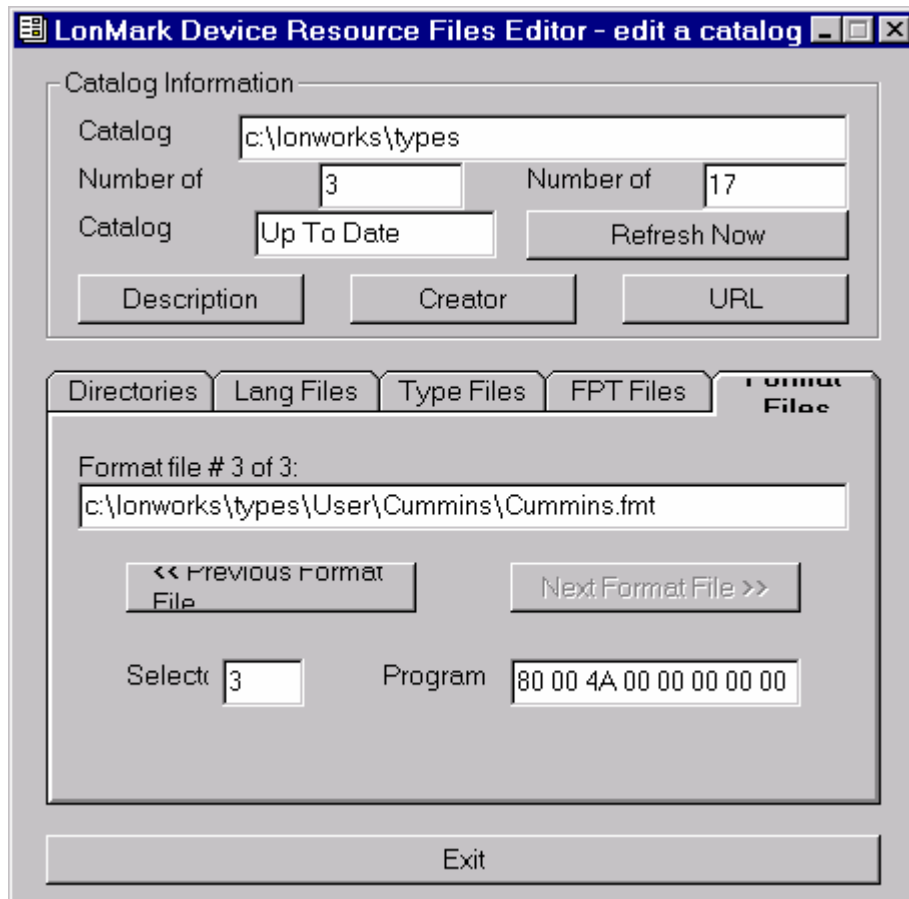
<< Previous Directory Next Directory >>

Remove This Directory

Add A New Directory

Exit

Step 1.7 Select the “Format Files” tab and confirm that “Cummins.fmt” is included by selecting the “Next Format File” button until “Cummins.fmt” appears. Do the same with “Type Files” and confirm that “Cummins.TYP” appears.

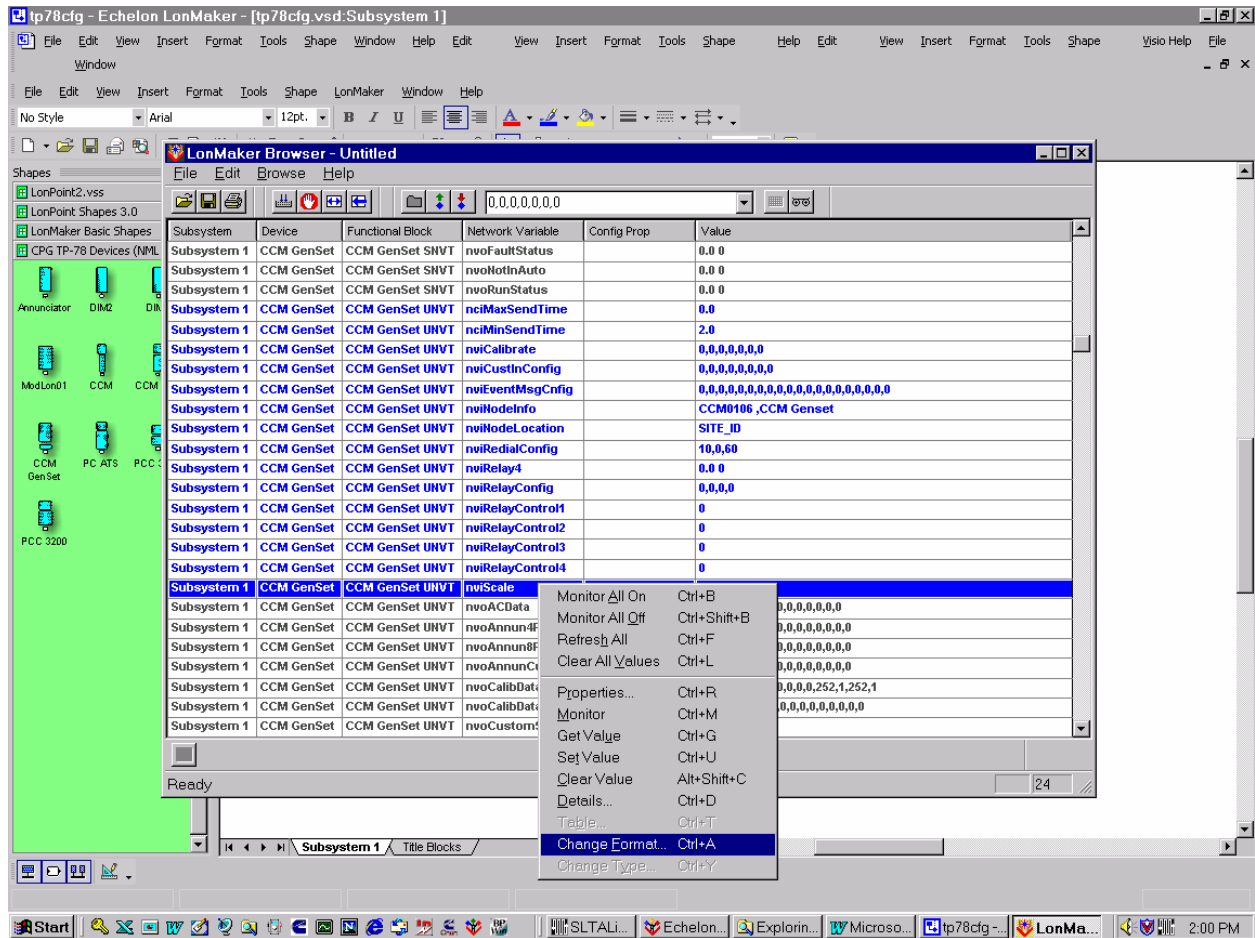


Step 2 Create the network with LonMaker for Windows.

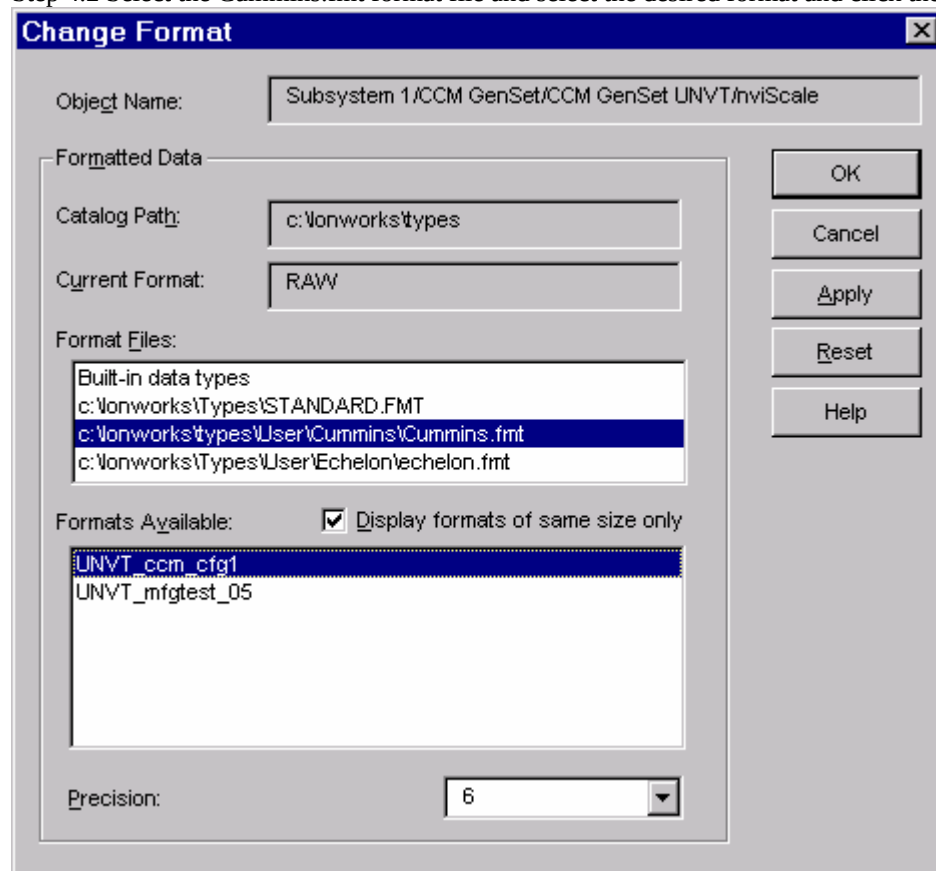
Step 3 Open the LonMaker Browser by right clicking on the device to be configured.

Step 4 Select variables to be modified, change format (optional) and modify the variables.

Step 4.1 To change format right click on the variable in the browser and select “Change Format”.



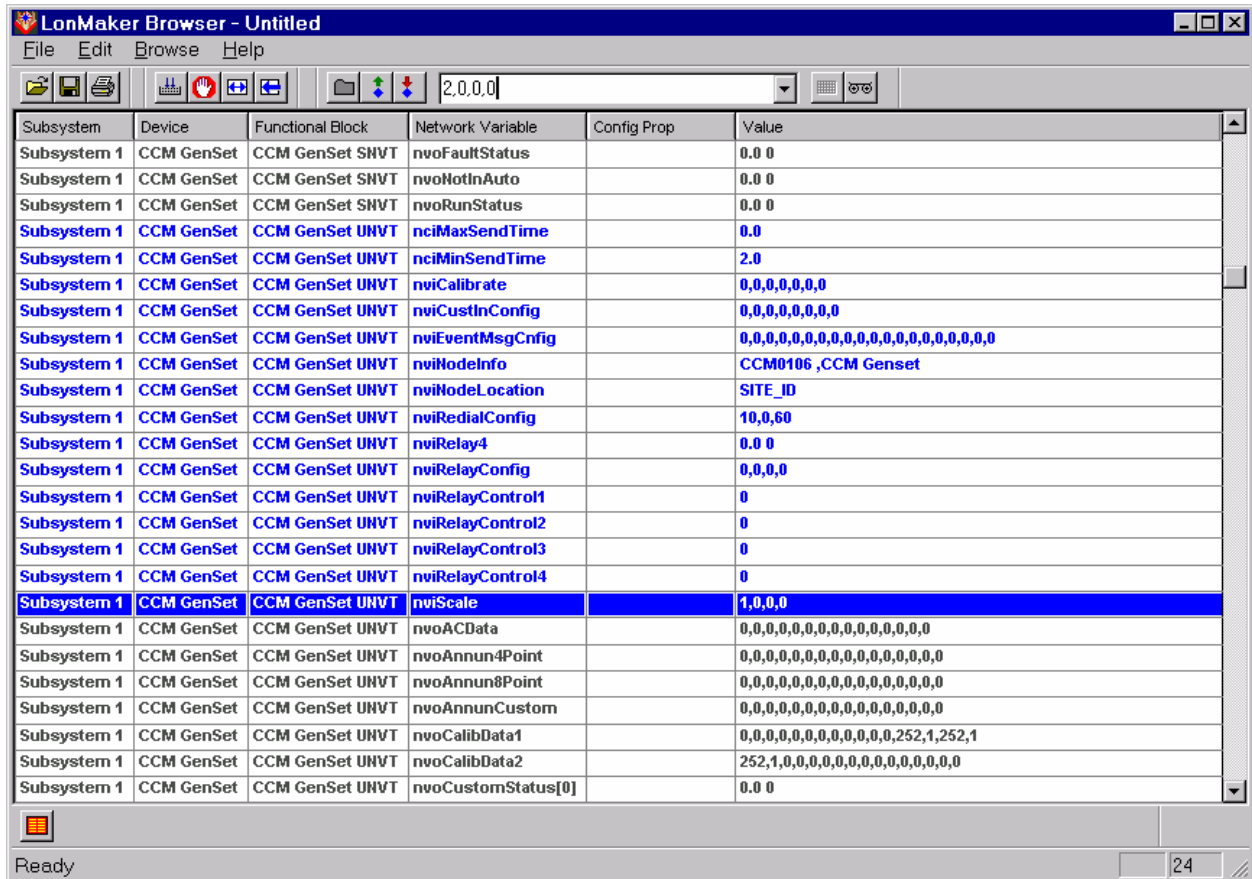
Step 4.2 Select the Cummins.fmt format file and select the desired format and click the “Apply” button.



The image shows a 'Change Format' dialog box with the following fields and controls:

- Object Name:** Subsystem 1/CCM GenSet/CCM GenSet UNVT/InviScale
- Formatted Data:** (Section header)
- Catalog Path:** c:\nonworks\types
- Current Format:** RAWV
- Format Files:** (List box containing:
 - Built-in data types
 - c:\nonworks\types\STANDARD.FMT
 - c:\nonworks\types\User\Cummins\Cummins.fmt (highlighted)
 - c:\nonworks\types\User\Echelon\echelon.fmt)
- Formats Available:** (List box containing:
 - UNVT_ccm_cfg1 (highlighted)
 - UNVT_mfgtest_05)
- Display formats of same size only:** ☒
- Precision:** 6 (dropdown menu)
- Buttons:** OK, Cancel, Apply, Reset, Help

Step 5 Change the variable by selecting it and then writing the new values for each of the fields in the text box at top of the window, and then clicking on the red down arrow icon to the left of the text box. All numbers should be rounded to the nearest whole number (no decimals).



Subsystem	Device	Functional Block	Network Variable	Config Prop	Value
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoFaultStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoNotInAuto		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoRunStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMaxSendTime		0.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMinSendTime		2.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCalibrate		0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCustInConfig		0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviEventMsgCnfig		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeInfo		CCM0106 ,CCM GenSet
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeLocation		SITE_ID
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRedialConfig		10,0,60
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelay4		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayConfig		0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl1		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl2		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl3		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl4		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviScale		1,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoACData		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun4Point		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun8Point		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnunCustom		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCalibData1		0,0,0,0,0,0,0,0,0,0,0,0,252,1,252,1
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCalibData2		252,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCustomStatus[0]		0.0 0

The following PCT parameters can be modified in this manner. Refer to Section 4 of the PowerCommand Network Installation and Operator's Manual (TP-78 version). Note that not all of the parameters set by PCT are written to the device. Some are written only to a .csv file to be used by PCW 1.

Module Name (p4-16) – Device name defined by the user in LonMaker.

Change the format of nviNodeInfor to UNVT_node_info. Write the following:

nviNodeInfo.name_tag = name

name_tag is the second field. If the first field has null characters in it (ascii 0) the second field may not show up in the browser. Change to raw data type, replace the nulls with spaces (ascii 32) and change back to UNVT_node_info. Note the above screen shot shows nviNodeInfo after the format has been changed to UNVT_node_info. If the raw data type is used, the variable will look like this:

Subsystem	Device	Functional Block	Network Variable	Config Prop	Value
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoFaultStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoNotInAuto		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoRunStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMaxSendTime		0.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMinSendTime		2.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCalibrate		0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCustInConfig		0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviEventMsgCnfig		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeInfo		67,67,77,48,49,48,54,32,67,67,77,32,71,101,110,115,101,116,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeLocation		SITE_ID
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRedialConfig		10,0,60
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelay4		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayConfig		0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl1		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl2		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl3		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl4		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviScale		1,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoACData		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun4Point		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun8Point		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnunCustom		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCalibData1		0,0,0,0,0,0,0,0,0,0,0,0,252,1,252,1
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCalibData2		252,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoCustomStatus[0]		0.0 0

Note that the 8th field has the number 32. This is the ASCII representation for a space. The browser may have spaces represented by the number 0, which is a null character. If any of the first 8 numbers are 0, change them to 32 before changing the format.

If raw data type is used then name tag is stored in bytes 8 – 23. Type in the ascii representation of each character.

Type – (p4-16) Not written to the device.

Site ID – (p4-16) Specified by the User when creating the site.

Change the format of nviNodeLocation to UNVT_node_loc. Write the following:

```
nviNodeLocation.site_id = (site id)
```

site id is the first field.

If raw data type is used then site id is stored in bytes 0 – 7. Type type in the ascii representation of each character.

Phase – (p4-16) Not written to the device

Voltage Scaling – (p4-17) This function accounts for external transformer turns ratios.

Change the format of nviScale to UNVT_ccm_cfg1. Write the following:

```
nviScale.Channel (byte 0)= 0
nviScale.Multiplier (bytes 1 and 2) = 236.6 * Phase A.PrimaryLN/Phase A Secondary LN
nviScale.Divider (bytes 3 and 4) = 512 (byte 3 = 2, byte 4 = 0)
nviScale.Offset (bytes 5 and 6) = 0
```

Repeat with Channel =1 for Phase B, and Channel =2 for phase C,

Channel is the first field, Multiplier is the second field, Divider is the third field and offset is the fourth field.

Note that to calculate the multiplier line-to-neutral values must be used. Divide line-to-line values by 1.732 to convert to line-to-neutral.

Example: (see example 1 from the TP-78 manual)

Primary voltage = 13,800 volts L-L
Secondary voltage = 120 volts L-N

Multiplier = $236.6 * (13,800 / 1.732) / 120 = 15,709$ (rounded to the nearest whole number)
Divider = 512
Offset = 0

Set Channel = 0 for phase A.

Enter 0,15709,512,0 for nviScale. See below.

Subsystem	Device	Functional Block	Set Value	Config Prop	Value
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nviShutdownCmd		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nviStartCmd		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoAnnunHFPA110		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoFaultStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoHotInAuto		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet SNVT	nvoRunStatus		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMaxSendTime		0.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nciMinSendTime		2.0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCalibrate		0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviCustInConfig		0,0,249,0,0,0,255,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviEventMsgCnfig		2,Common Alarm ,17
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeInfo		CCM0106 ,CCM GenSet
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviNodeLocation		SITE_ID
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRedialConfig		10,60
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelay4		0.0 0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayConfig		0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl1		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl2		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl3		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviRelayControl4		0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nviScale		0,15709,512,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoACData		0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun4Point		0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
Subsystem 1	CCM GenSet	CCM GenSet UNVT	nvoAnnun8Point		0,0

Set value of selected object

Repeat setting channel = 1 for Phase B and 2 for Phase C.

If raw data type is used then Byte 0 is the channel, bytes 1 and 2 are the multiplier, 3 and 4 the divider and 5 and 6 the offset. Bytes 1, 3 and 5 are the most significant bytes and bytes 2, 4 and 6 are the least significant bytes. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 513, the most significant byte is 2 and the least significant byte is 1. ($513/256 = 2$ with a remainder of 1.)

Current Scaling – (p4-18) This function accounts for current transformer turns ratios.

Change the format of nviScale to UNVT_ccm_cfg1. Write the following:

```
nviScale.Channel (byte 0)= 3
nviScale.Multiplier (bytes 1 and 2) = 6.104 * Phase A.Primary/Phase A Secondary
nviScale.Divider (bytes 3 and 4) = 1023 (byte 3 = 3, byte 4 = 255)
nviScale.Offset (bytes 5 and 6) = 0
```

Repeat with Channel =4 for Phase B, and Channel =5 for phase C,

Channel is the first field, Multiplier is the second field, Divider is the third field and offset is the fourth field.

If raw data type is used then Byte 0 is the channel, bytes 1 and 2 are the multiplier, 3 and 4 the divider and 5 and 6 the offset. Bytes 1, 3 and 5 are the most significant bytes and bytes 2, 4 and 6 are the least significant bytes. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 1023, the most significant byte is 3 and the least significant byte is 255. ($1023/256 = 3$ with a remainder of 255.)

Relays – (p4-19) Configures CCM relays to be pulsed or maintained. (“Generic” CCM only)

Change the format of nviScale to UNVT_ccm_cfg3. Each of the four fields corresponds to one relay. For each relay, write 1 for maintained, 2 for pulsed.

Dial Out – (p4-20) This configures redial operation.

Change the format of nviScale to UNVT_node_redial. Write the following:

```
nviRedialConfig.attempts (byte 0)
nviRedialConfig.delay (bytes 1 and 2)
```

Attempts is the first field and delay is the second field.

If raw data type is used then Byte 0 is the attempts and bytes 1 and 2 are the delay. Byte 1 is the most significant byte and byte 2 is the least significant byte. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 1023, the most significant byte is 3 and the least significant byte is 255. ($1023/256 = 3$ with a remainder of 255.)

Analog Inputs – scaling of analog channels

Temperature 1, 2, 3 and Oil Pressure – (p4-20)

Change the format of nviScale to UNVT_ccm_cfg1. Write the following:

```
nviScale.Channel (byte 0)= 9
nviScale.Multiplier (bytes 1 and 2) = 10.01 *Sender Gain
```

nviScale.Divider (bytes 3 and 4) = 1023 (byte 3 = 3, byte 4 = 255)
nviScale.Offset (bytes 5 and 6) = (Meter Reading / Sender Gain – Sender Voltage) * 1023/10.01

Repeat with Channel = 10 for Temperature 2, 11 for Temperature 3 and 12 for Oil Pressure

Channel is the first field, Multiplier is the second field, Divider is the third field and offset is the fourth field.

If raw data type is used then Byte 0 is the channel, bytes 1 and 2 are the multiplier, 3 and 4 the divider and 5 and 6 the offset. Bytes 1, 3 and 5 are the most significant bytes and bytes 2, 4 and 6 are the least significant bytes. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 1023, the most significant byte is 3 and the least significant byte is 255. ($1023/256 = 3$ with a remainder of 255.)

Spare 1 and spare 2 – (p4-23)

Change the format of nviScale to UNVT_ccm_cfg1. Write the following:

nviScale.Channel (byte 0)= 13
nviScale.Multiplier (bytes 1 and 2) = 65.04 *Sender Gain
nviScale.Divider (bytes 3 and 4) = 1023 (byte 3 = 3, byte 4 = 255)
nviScale.Offset (bytes 5 and 6) = (Meter Reading / Sender Gain – Sender Voltage) * 1023/6.504

Repeat with Channel = 14 for spare 2

Channel is the first field, Multiplier is the second field, Divider is the third field and offset is the fourth field.

If raw data type is used then Byte 0 is the channel, bytes 1 and 2 are the multiplier, 3 and 4 the divider and 5 and 6 the offset. Bytes 1, 3 and 5 are the most significant bytes and bytes 2, 4 and 6 are the least significant bytes. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 1023, the most significant byte is 3 and the least significant byte is 255. ($1023/256 = 3$ with a remainder of 255.)

Spare 3 – (p4-23)

Change the format of nviScale to UNVT_ccm_cfg1. Write the following:

nviScale.Channel (byte 0)= 15
nviScale.Multiplier (bytes 1 and 2) = 384.2 *Sender Gain
nviScale.Divider (bytes 3 and 4) = 1023 (byte 3 = 3, byte 4 = 255)
nviScale.Offset (bytes 5 and 6) = (Meter Reading / Sender Gain – Sender Voltage) * 1023/38.42

Channel is the first field, Multiplier is the second field, Divider is the third field and offset is the fourth field.

If raw data type is used then Byte 0 is the channel, bytes 1 and 2 are the multiplier, 3 and 4 the divider and 5 and 6 the offset. Bytes 1, 3 and 5 are the most significant bytes and bytes 2, 4 and 6 are the least significant bytes. The most significant byte of a two byte number is calculated by dividing the number by 256 and discarding the remainder. The least significant byte is the remainder. For example, if the number is 1023, the most significant byte is 3 and the least significant byte is 255. ($1023/256 = 3$ with a remainder of 255.)

Discrete Inputs: - (p4-24) This defines Event type, active state, event text and notify setting for each of the 32 discrete inputs.

Change the format of nviCustInConfig to UNVT_ccm_cfg2. Change the format of nviEventMsgConfig to UNVT_ccm_cfg3. Write the following:

nviCustInConfig.active[0] = Active state of Inputs 1 – 8 (1 is LSB) Closed = 0, Open = 1
 nviCustInConfig.active[1] = Active state of Inputs 9 – 16 (9 is LSB) Closed = 0, Open = 1
 nviCustInConfig.active[2] = Active state of Inputs 17 – 24 (17 is LSB) Closed = 0, Open = 1
 nviCustInConfig.active[3] = Active state of Inputs 25 – 32 (25 is LSB) Closed = 0, Open = 1
 nviCustInConfig.notify[0] = Notify Setting of Inputs 1 – 8 (1 is LSB) No = 0, Yes = 1
 nviCustInConfig.notify[1] = Notify Setting of Inputs 9 - 16 (9 is LSB) No = 0, Yes = 1
 nviCustInConfig.notify[2] = Notify Setting of Inputs 17 - 24 (17 is LSB) No = 0, Yes = 1
 nviCustInConfig.notify[3] = Notify Setting of Inputs 25 –32 (25 is LSB) No = 0, Yes = 1
 nviEventMsgConfig.fault_type: 0 for Event, 1 for Warning, 2 for Shutdown
 nviEventMsgConfig.fault_text[0..15]: Any 16 character string
 nviEventMsgConfig.input = Input number.

For nviCustInConfig active[0] through active[3] are fields 1–4 , notify[0] through notify[3] are fields 5 – 8. For nviEventMsgConfig fault_type is the first field, fault_text consists of the next 16 characters and input is the last field.

As an example, see p4-24 of the TP-78 manual, where inputs 17-24 are configured. Inputs 18 and 19 are configured to be active closed and the rest are active open. The active state of inputs 17-24 are set by the third field of nviCustInConfig, with Input 24 controlled by the most significant bit in that field and input 17 controlled by the least significant bit. To calculate the value to insert in that field use the following formula:

$$\begin{aligned}
 \text{nviCustInConfig.active[2]} = & 128 * \text{Input 24} \\
 & + 64 * \text{Input 23} \\
 & + 32 * \text{Input 22} \\
 & + 16 * \text{Input 21} \\
 & + 8 * \text{Input 20} \\
 & + 4 * \text{Input 19} \\
 & + 2 * \text{Input 18} \\
 & + 1 * \text{Input 17}
 \end{aligned}$$

where Input X = 1 if that input is active open and 0 if it is active closed. In this example, inputs 18 and 19 are active closed and the rest are active open. Therefore

$$\text{nviCustInConfig.active[2]} = 128 + 64 + 32 + 16 + 8 + 0 + 0 + 1 = 249$$

Similarly, the notify bits for these 8 inputs are in field 7. The bit should be set to 1 if an alarms is to be sent when this event is active and a 0 otherwise. In this example, all 8 inputs are set for notify

$$\text{nviCustInConfig.notify[2]} = 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255$$

For event text, nviEventMsgConfig must be written for each event, so to set up event text for 16 inputs you must write to nviEventMsgConfig 16 times. In the example input 17 is configured as a shutdown, with “Common Alarm” as the text. The screen shot on page 9 illustrates how this should be written. Notice that there are trailing spaces after “Common Alarm”. This field must have 16 characters, so spaces need to be added to any shorter text strings.

If raw data type is used for nviEventMsgConfig , bytes 0 and 1 are fault type, bytes 2 – 17 are fault text and bytes 18 and 19 are input number.

Notes

Module name, Site ID, dialout attempts and redial delay are all configured for the GCM in the same way as they are for the CCM.

Configure the NGM (SLTA-10) using LinkManager.

