

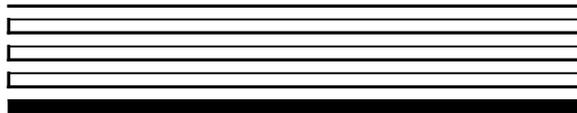
Alarm Dialer User's Guide

Version 1



Corporation

005-0134-01A



Echelon, LON, LonTalk, LonBuilder, NodeBuilder, Neuron, LonManager, LONMARK, LONWORKS, 3120 and 3150 are registered trademarks of Echelon Corporation. LonMaker is a trademark. Other brand and product names are trademarks or registered trademarks of their respective holders.

Neuron Chips, Serial LonTalk Adapters, and other OEM Products were not designed for use in equipment or systems which involve danger to human health or safety or a risk of property damage and Echelon assumes no responsibility or liability for use of these products in such applications.

Parts manufactured by vendors other than Echelon and referenced in this document have been described for illustrative purposes only, and may not have been tested by Echelon. It is the responsibility of the customer to determine the suitability of these parts for each application.

ECHELON MAKES AND YOU RECEIVE NO WARRANTIES OR CONDITIONS, EXPRESS, IMPLIED, STATUTORY OR IN ANY COMMUNICATION WITH YOU, AND ECHELON SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Echelon Corporation.

Printed in the United States of America.
Copyright ©1999 by Echelon Corporation.

Echelon Corporation
4015 Miranda Avenue
Palo Alto, CA 94304, USA
www.echelon.com

Contents

1 Alarm Dialer Overview and General Setup	1-1
Introduction	1-2
Related Documentation	1-2
Software Installation	1-2
General Operation	1-3
Hardware Components	1-3
Software Components	1-4
Hardware Setup	1-4
Creating the LonMaker Drawing	1-7
The Finished Network	1-13
2 Alarm Dialer Installation Scenarios	2-1
Scenario #1: Local Hardware Fault Alarm	2-2
Scenario #2: Remote Hardware Fault	2-10
DM-20/21 Event-based Alarm	2-11
3 Functional Block & Plug-in Reference	3-1
Functional Block Overview	3-2
Virtual Functional Block	3-2
Node Object	3-3
Digital Inputs	3-3
Digital Encoder	3-4
DM Dialer	3-4
Plug-in Reference	3-5
Appendix A - Plug-in Response Messages	A-1
Informational Messages	A-2
Error Messages	A-2
Appendix B - Data Structures	B-1
Alarm Dialer CP Structures	B-2

1

Alarm Dialer Overview & General Setup

This section provides an overview of the Alarm Dialer features, and
Describes the basic hardware and software setup required to use the
Alarm Dialer.

Introduction

The Alarm Dialer is an event driven system that sends numeric pages based on network events. Network service personnel can receive a page notification of a problem directly from the network – no PC is required on site. Network events can originate from three sources: a direct digital input, a specific DM20/21 event, and network variable inputs.

The Alarm Dialer software runs on a standard LonPoint Digital Input (DI) module. You must overwrite the default application image of the DI with the Alarm Dialer software. The DI is then dedicated to alarm processing. The Alarm Dialer software monitors network events, decides which events constitute an alarm and uses an SLTA-10 with compatible modem to send the numeric pages.

Alarm events are defined/customized using the Alarm Dialer Plug-in.

Related Documentation

The Alarm Dialer relies on four hardware components to perform its task: a LonPoint Digital Input module, an SLTA-10, a Hayes compatible modem, and optionally the DM 20/21 Device Manager. The following manuals are suggested reading for details on setting up each component.

- LonPoint Digital Input Interface Module
 - LonPoint System Overview, available from Echelon's website (<http://www.echelon.com/Products/lonpoint/Default.htm>)
 - DI-10 Datasheet, available from Echelon's website (<http://ni.echelon.com/datasheets/di10.pdf>)
- SLTA-10 Serial LonTalk® Adapter
 - Serial LonTalk Adapter Datasheet, available from Echelon's website (http://www.echelon.com/Products/Net_Interface_Prods/7335x.pdf)
 - SLTA-10 Adapter User's Guide, available from Echelon's website (<http://www.echelon.com/products/technical/pdfs/manuals/slta-10.pdf>)
- Device Manager
 - DM-20/21 User's Guide, available from Echelon's website (<http://www.echelon.com/products/dm>)

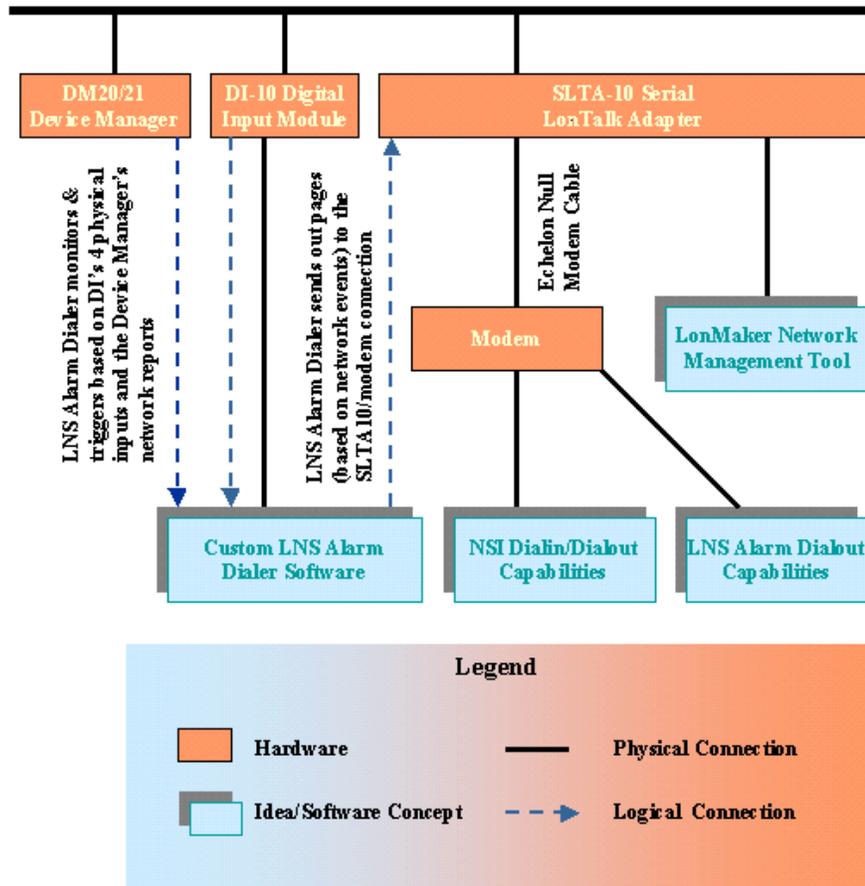
Software Installation

Before you install the Alarm Dialer software, be sure that the LonMaker for Windows and the SLTA link manager software and (any other LonWorks software you will be using) is installed in a designated LonWorks directory.

To install the Alarm Dialer software, insert the CD into your CD-ROM drive. Navigate to the CD-ROM drive and double-click on Setup.exe. Follow the setup wizard's on screen instructions.

You must install the network using the SLTA as the LonMaker network interface. It is not possible to configure the alarm dialer with any other network interface.

General Operation



If an alarm dial-out condition is reached, the Alarm Dialer software (running on the Digital Input module) will direct the SLTA-10 to dial the number of your paging service. It will then direct the SLTA-10 to send the DTMF tones required to execute a page.

Hardware Components

The Alarm Dialer requires the following devices to be present on the network:

- LonPoint Digital Input Interface Module (DI-10 Model 41100)
- SLTA-10 Serial LonTalk® Adapter (Models 73351, 73352, 73353, or 73354)
- Hayes compatible modem

To use the “DM Dialer” feature of the Alarm Dialer, a Device Manager (DM-20, model 43201 / DM-21, model 43202) module must also be present.

Software Components

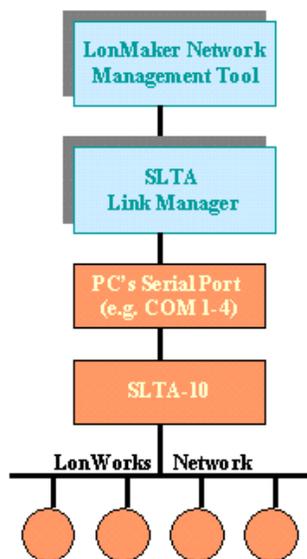
The Alarm Dialer software is made up of two components:

1. Alarm Dialer firmware (loadable into a standard DI module)
2. Alarm Dialer plug-in

Both of these components are installed as part of the Alarm Dialer software installation.

Hardware Setup

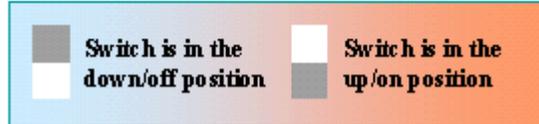
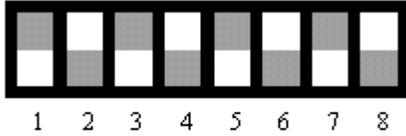
Setup At Install Time



1. Attach a LonPoint Digital Input module to the network. (Consult the Echelon LonPoint User's Guide for more details.)
2. Attach a Serial LonTalk Adapter (SLTA-10) to the network. (See SLTA-10 Adapter User's Guide for more details)

The configuration dip-switches on the SLTA-10 must be set to the following positions. These settings allow the Alarm Dialer software to dial-out and allow the SLTA-10 to function in NSI mode for remote network monitoring. The last three configuration switches may be changed to accommodate different modem speeds. For more details on the SLTA-10 configuration switch settings, please consult Chapter 4 of the SLTA-10 Adapter User's Guide.

SLTA-10 Configuration Switch Settings



Switch #	Up/On Position	Down/Off Position	Set To
1	Buffered Link Protocol	ALERT/ACK Link Protocol	Down/Off
2	Remote Host (modem)	Local Host (default)	Up/On
3	Network Disable On	Network Enable (Disable off)	Down/Off
4	NSI Mode Firmware	MIP Mode Firmware	Up/On
5	Autobaud Enabled	Autobaud Disabled	Down/Off
6-8	See next table	See next table	User Defined

Dip Switch Settings for SLTA-10 Baud Rate Control

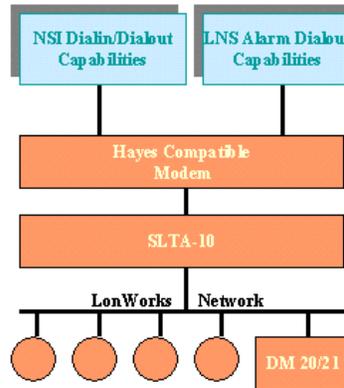
Switch 6	Switch 7	Switch 8	Selected Baud Rate
On	On	On	115,200 bps
On	On	Off	57,600 bps
On	Off	On	38,400 bps
On	Off	Off	19,200 bps
Off	On	On	9600 bps
Off	On	Off	2400 bps
Off	Off	On	1200 bps
Off	Off	Off	300 bps

Note: in order for switch 6, 7, & 8 to take effect, the SLTA-10's *autobaud* feature must be disabled.

3. Connect the SLTA-10's EIA-232 port to the PC's serial port. Note that the PC must have LonMaker (or other compatible director application) installed to run the Alarm Dialer plug-in. ***The SLTA-10 must be used as the network interface for LonMaker, and will be represented as the "LNS Network Interface" device in the LonMaker drawing, during the installation phase.***
4. Optionally, if triggering alarm conditions based on Device Manager events is desired, attach a DM20/21 device to the network. Use the DM loading wizard to load it with the appropriate network database. If a DM-20/21 is attached to the network it is responsible for commissioning all devices on the network. See the DM-20/21 user's guide for details. ***The following steps assume that a DM-20/21 is not attached, and that "device commissioning" is done by LonMaker for Windows.***

- Once the default DI software has been overwritten by the Alarm Dialer .APB file, and all devices have been commissioned, the SLTA-10's EIA-232 connection should be removed from the PC and connected to the modem using a null modem cable. The modem and SLTA-10 must remain on site, but the PC may be removed. (Device commissioning is explained in the next section)
- Attach the modem to a power supply and a phone line. The phone line should be dedicated to the Alarm Dialer's dial-out, and the NSI mode remote network monitoring dial-in.

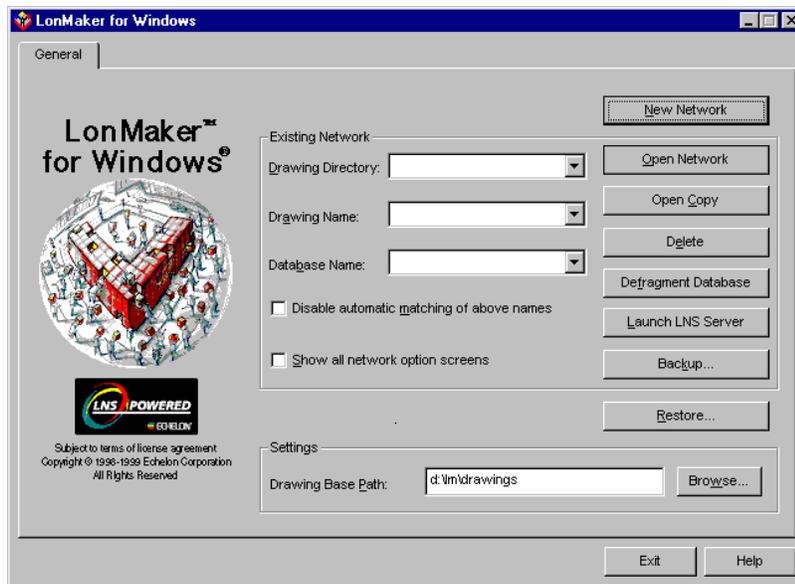
**System Left On-Site After Installation
(with DM-based Alarm Conditions)**



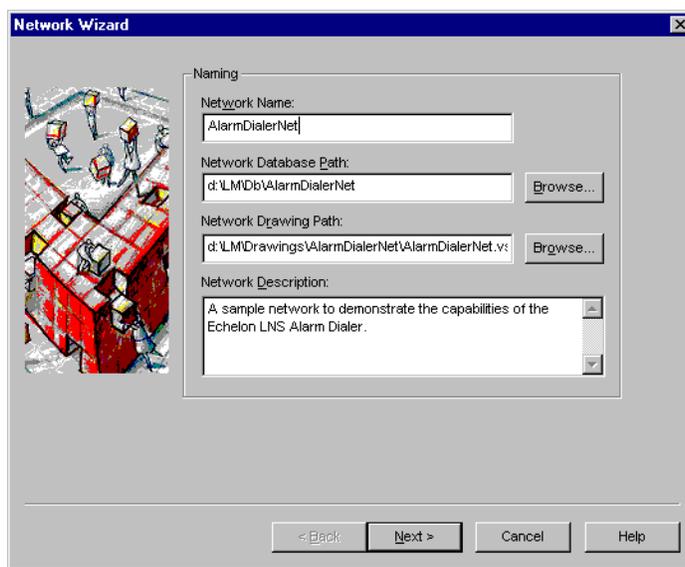
Creating the LonMaker Drawing

The next section in this manual describes three different Alarm Dialer scenarios. Each of the three scenarios builds upon this basic LonMaker drawing.

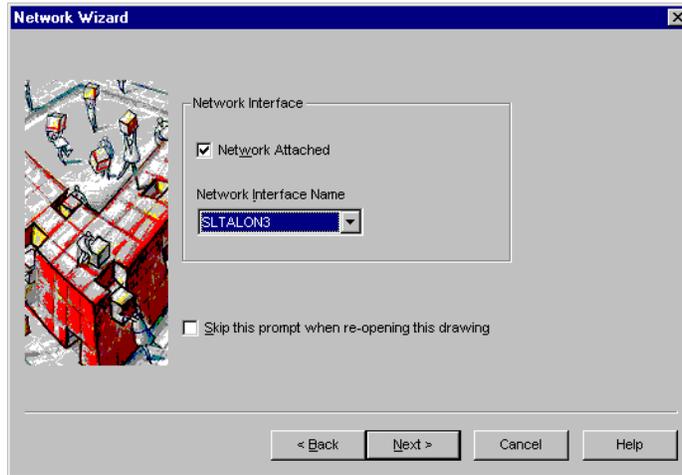
1. Run the LonMaker software. The main LonMaker screen will appear. Select *New Network* to create a new LonMaker drawing..



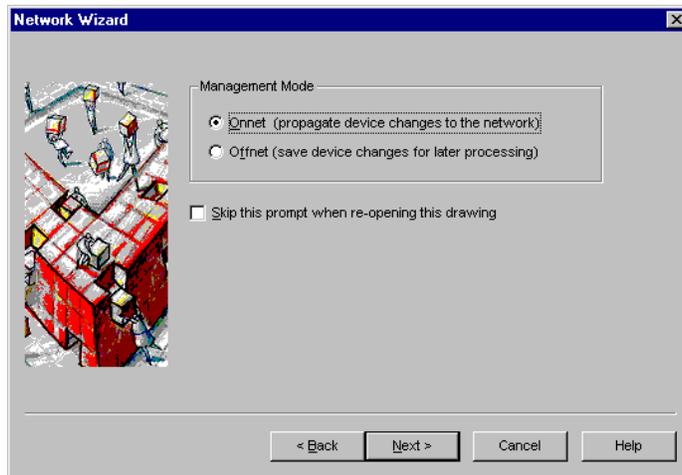
2. Type in the desired network name and an optional network description. Click *Next* to continue.



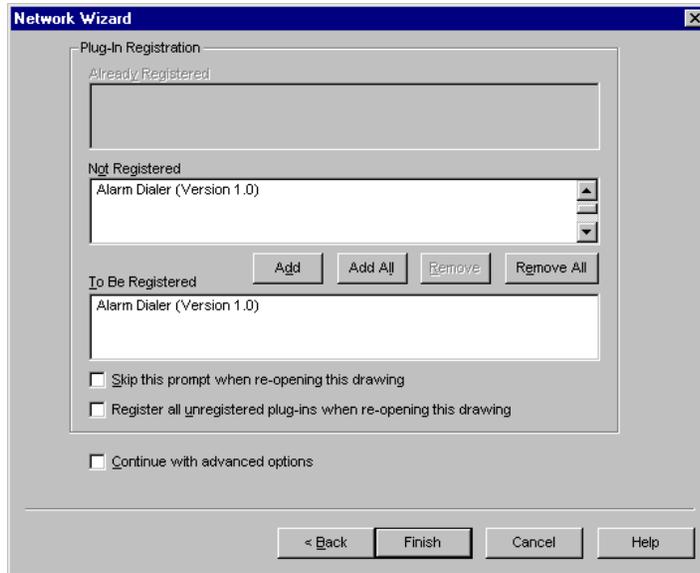
3. Select the network interface that corresponds to the SLTA-10. Be sure to check *Network Attached*, and then Click *Next* to continue.



4. Open the network in *On-Net* mode, and click *Next* to continue.

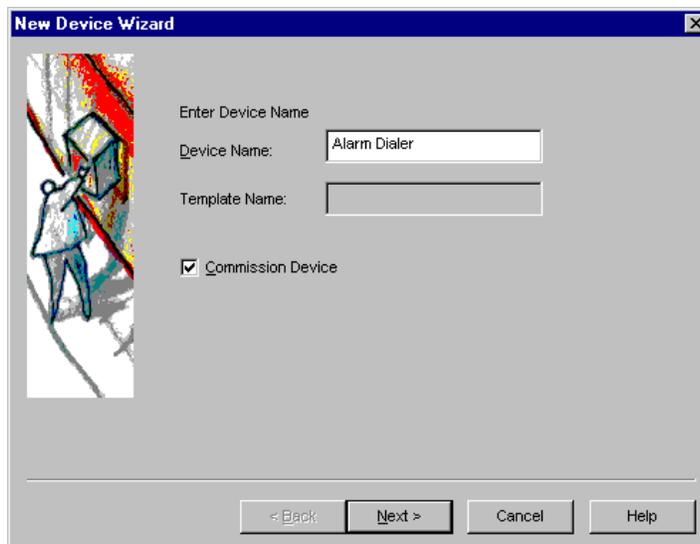


5. Select Alarm Dialer (Version 1.0) in the Not Registered pane. Click on *Add* to have LonMaker register the Alarm Dialer plug-in for use in the network. Be sure the SLTA-10 Link Manager is running in the background with a successful connection, and click *Finish* to continue.

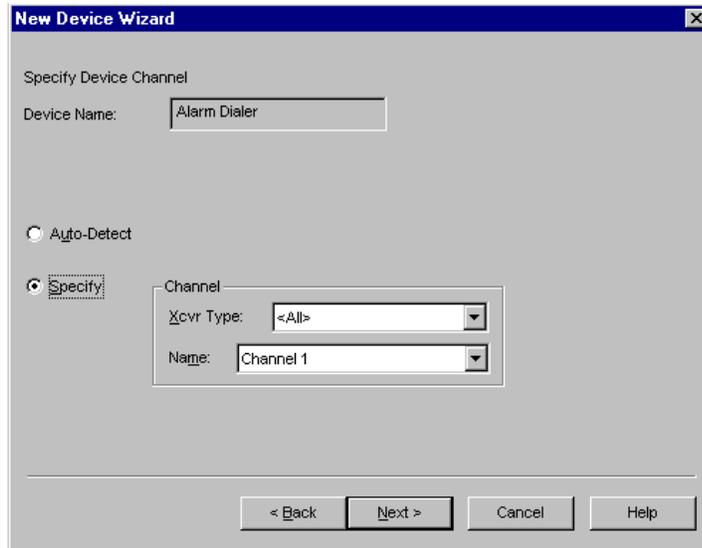


LonMaker will now display a Visio drawing page. The local network interface (SLTA-10) and default channel will be displayed on the drawing. Next we will add components to the network.

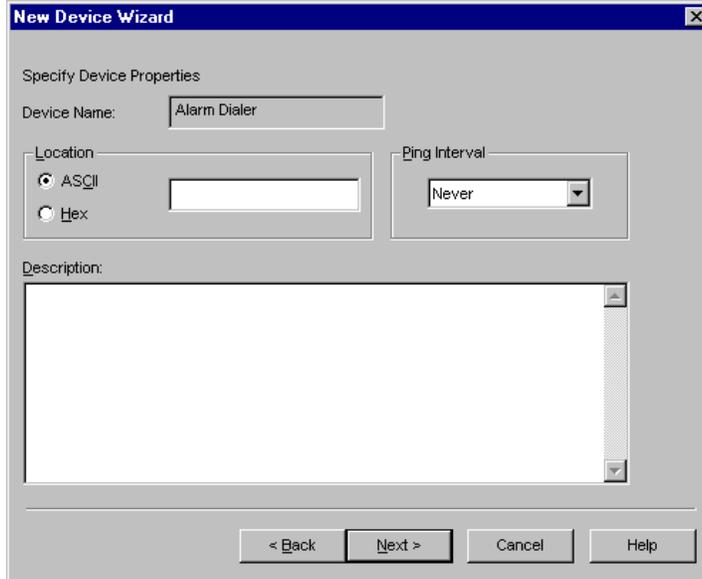
6. Select *Alarm Dialer*. From the from the LonMaker *File | Stencils* menu. A new stencil will now appear. Select *Alarm Dialer* from the Alarm Dialer stencil, and drag it onto your drawing. The “New Device Wizard” window will appear. Type in a name that will represent the Alarm Dialer module on the network, and check the “Commission Device” box. Click *Next* to continue.



7. Select the *Specify* checkbox and choose the desired transceiver type & channel. Click *Next* to continue.



8. Type in an optional location string and description for the Alarm Dialer DI. Select *Never* for the Ping Interval, and click *Next* to continue.



9. Select the *service pin* method for identifying the Alarm Dialer DI, and click *Next* to continue.

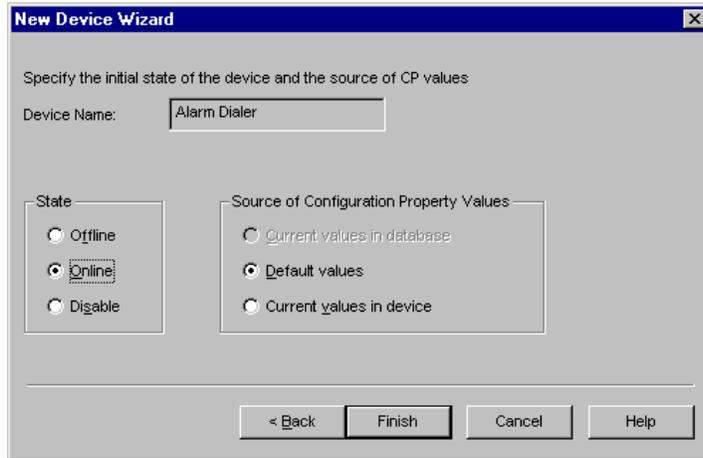
The screenshot shows the 'New Device Wizard' dialog box with the title bar 'New Device Wizard'. The main area is titled 'Identify Device'. There is a text box for 'Device Name' containing 'Alarm Dialer'. Below this, there are two radio button options: 'Service Pin' (which is selected) and 'Manual'. To the right of the 'Manual' option is a text box for 'Neuron ID'. At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

10. ***This step instructs LonMaker to load the custom alarm processing firmware into the DI node.*** Check the *Load Application Image* box. Point to the AlarmDialer.APB (the application image file) and AlarmDialer.xif (the external interface file). These files can be found under the \LonWorks\Import\Echelon\Alarm Dialer\ directory. Click *Next* to continue.

The screenshot shows the 'New Device Wizard' dialog box with the title bar 'New Device Wizard'. The main area is titled 'Specify device application image name'. There is a text box for 'Device Name' containing 'Alarm Dialer'. Below this, there is a checked checkbox labeled 'Load Application Image'. Underneath, there are two text boxes: 'Image Name' containing 'd:\lonworks\import\Echelon\Alarm Dialer\AlarmDialer.APB' and 'XIF Name' containing 'd:\lonworks\import\Echelon\Alarm Dialer\AlarmDialer.XIF'. Each text box has a 'Browse...' button to its right. At the bottom of the dialog, there are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

The DI will no longer function as a DI, but will instead function as the Alarm Dialer. This should be noted on the physical device by removing any sticker indicating the device is a DI-10, and labeling it as the “Alarm Dialer” instead.

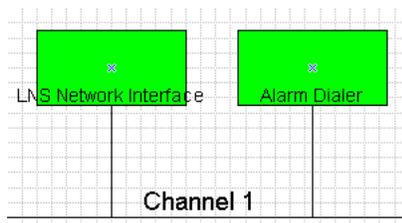
- Select the *Online* state for the device, and use the *Default Values* configuration property values. Click *Finish* to continue.



- Place a checkmark by *Filter on program ID* and press the service pin on the Digital Input module.



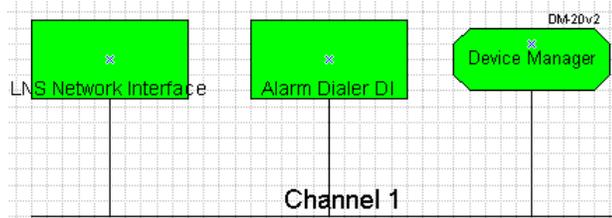
- LonMaker will load the Alarm Dialer application and commission the DI in a single step. At this point, two devices should be configured on the network – the SLTA-10 used as the LNS network interface, and the Alarm Dialer module.



If triggering alarm conditions based on Device Manager events is desired, a DM-20 or DM-21 must be included in the LonMaker drawing. See the DM-20/21 User's guide for detailed information on using the device manager.

The Finished Network

The completed LonMaker drawing should contain the following devices. Note that the "LNS Network Interface" is actually the SLTA-10 that will remain on site, craftily used for a network interface in LonMaker.



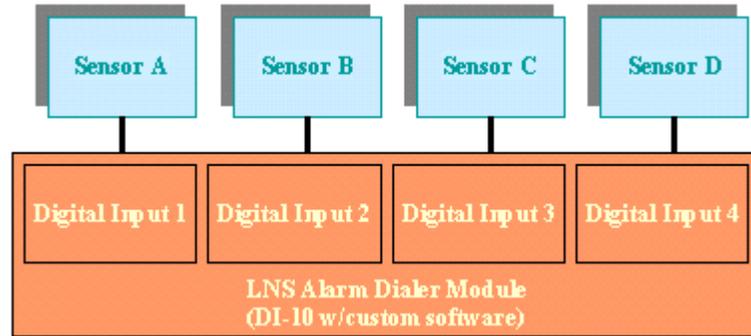
2

Alarm Dialer Installation Scenarios

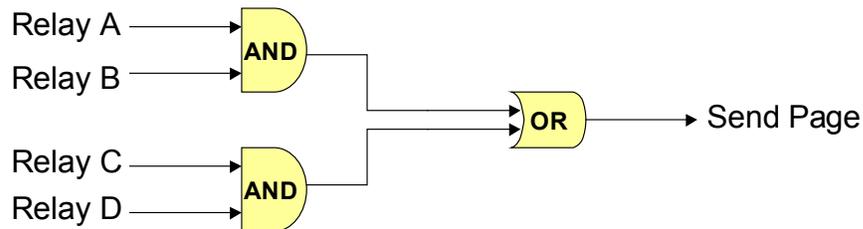
This section covers three installation scenarios. It assumes a LonMaker drawing has been created as described in the previous section.

Scenario #1: Local Hardware Fault Alarm

The Alarm Dialer's four physical inputs are connected to relays. An alarm condition occurs when relays A & B close or when relays C & D close. If an alarm condition is reached, a page is sent to the numeric pager.



Scenario 1 Setup

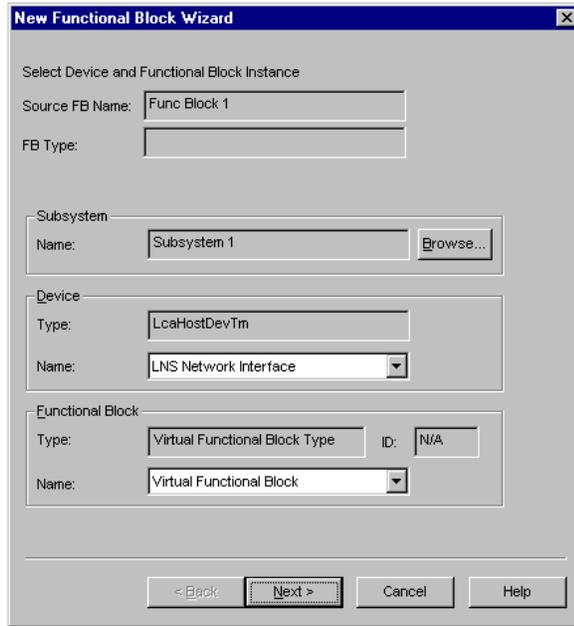


Scenario 1 Alarm Logic

Setting up Scenario 1 in LonMaker

1. Create a base LonMaker drawing as described in chapter 1 that contains the SLTA-10 and DI node. The DI should be loaded with the Alarm Dialer application.
2. Open the "Echelon Alarm Dialer" stencil set. (If the stencil set is not visible, from the "File" menu, select "Stencils" and then choose "Echelon Alarm Dialer.")

1. Add the SLTA-10's "Virtual Functional Block" to the network. From the "LonMaker Basic Shapes" stencil set, select the "Functional Block" icon and drag it onto the network. The "New Functional Block Wizard" will appear. From the "Device Name" dropdown box, select "LNS Network Interface." From the "Functional Block" dropdown box, select "Virtual Functional Block." Click **Next** to continue.

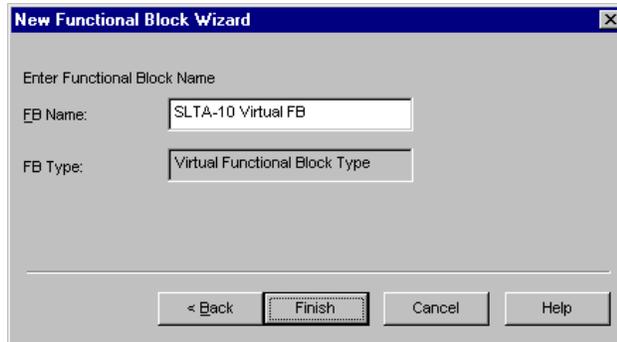


Notice that you are adding the Virtual Functional Block of the "LNS Network Interface". LonMaker doesn't realize that you are using an SLTA-10 as your network interface, so the best description it can give is "LNS Network Interface".

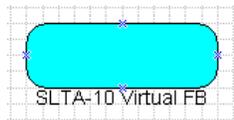
For the Virtual Functional Block to actually correspond to an SLTA-10 you must be using an SLTA-10 to connect the LonMaker PC to the network. (e.g. you must commission all the devices in the network through the SLTA-10) This is critical! If you use any other type of network interface the alarm dialer will not work.

The reason for this is that you will ultimately leave the SLTA-10 behind so that it can be used by the network to initiate a dial-out telephone call, and unlike most applications, the Alarm dialer binds to the SLTA-10. If you were using a network interface that were removed when the PC was disconnected (like a PCC-10) you would physically break the bind. The SLTA-10 is the only network interface that can be powered without a PC and therefore left on the job site. Remember that the end goal here is to create a system that can dial a pager without leaving the PC on site.

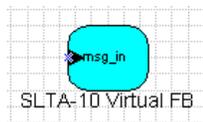
2. Type an appropriate name for the SLTA-10's Virtual Functional Block, and Click Finish to continue.



3. A Virtual Functional Block from the SLTA-10 device will appear on the network drawing.

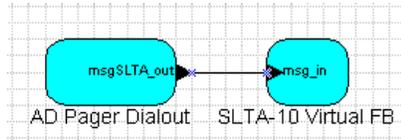


4. From the "LonMaker Basic Shapes" stencil set, select the "msg_in" icon, and drag it onto the functional block created in step 3. When the "Choose a Message Tag" screen appears, select "msg_in" and click on the "OK" button. The functional block, will now contain an input message tag.

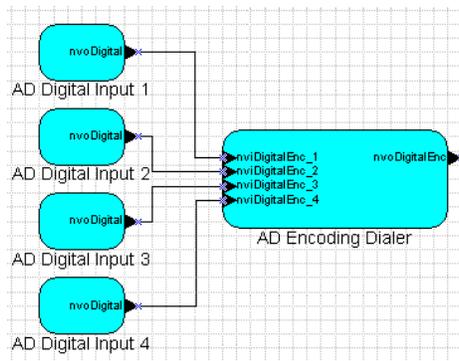


5. From the "Echelon Alarm Dialer" stencil set, select the "Pager Dialout" icon, and drop it onto the network. Repeat steps 1 & 2 when the "New Functional Block Wizard" appears. At this point, two functional blocks should exist in the drawing.

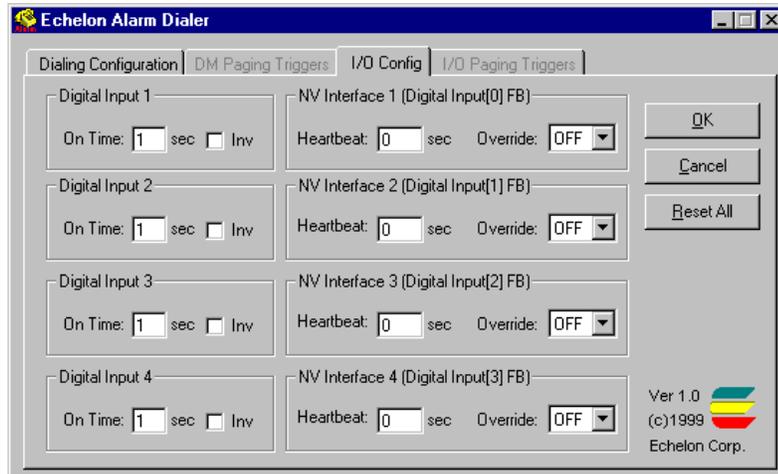
- Using the “Connector” icon from the “LonMaker Basic Shapes” stencil, bind output from the “AD Pager Dialout” FB to the input of the SLTA-10 Virtual Functional Block.



- From the “Echelon Alarm Dialer” stencil set, drop the four digital input functional blocks onto the network. (“Digital Input 1” through “Digital Input 4”).
- From the “Echelon Alarm Dialer” stencil set, drop the “Encoding Dialer” functional block onto the network.
- Connect the “Digital Input” functional blocks to the appropriate inputs on the “Encoding Dialer” functional block.



- Right-click on one of the “Digital Input” functional blocks, and select “Configure”. The Alarm Dialer plug-in will pop up with the “I/O Config” tab selected. Digital input settings are configured using this tab.

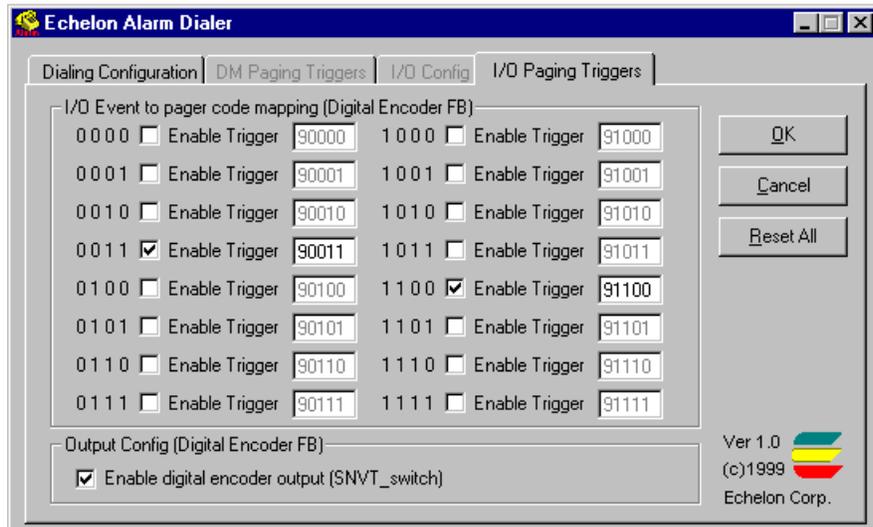


- “On Time” selects how many seconds the signal must remain HIGH or LOW before the value will be sent on the “nvoDigital” output network variable. A 100ms debounce time is always included, and therefore the total wait is 100ms + ‘On Time’.
- “Inv” (or Invert) selects which signal state constitutes a TRUE output on the “nvoDigital” output network variable. Leaving this box unchecked means that the input is active high, checking this box means the signal is active low.
- The “Heartbeat” value selects how often the “nvoDigital” variable is updated. Choose 0 seconds to send updates only when the physical input state changes.
- “Override” selects the “nvoDigital” value sent while the functional block is in override mode. See the “LonMaker Plug-in Reference” section later in this manual for more details.

In this scenario, a one-second “On Time” will be used. The inputs do not need to be inverted and heartbeats & override values are not used.

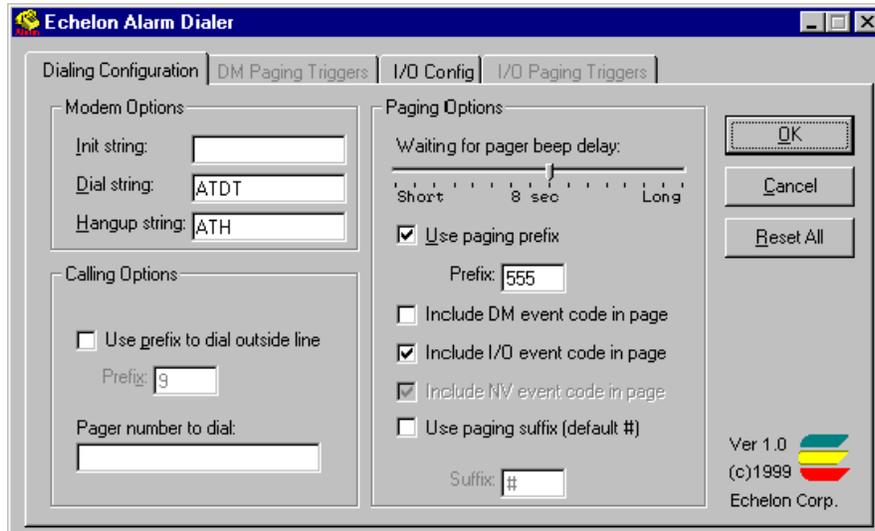
Click on “OK” once all options are set to return to your LonMaker drawing.

11. Right-click on the “Encoding Dialer” Functional block, and select “Configure.” The Alarm Dialer plug-in will pop up with the “I/O Paging Triggers” tab selected. Using this tab, you can configure which combination of physical inputs constitutes an alarm, and which numeric pager string to send in each case. In this scenario, the tab should be configured as follows:



- The “0 0 0 0” through “1 1 1 1” labels represent possible physical input states, as represented in the “Scenario 1 Alarm Logic.” The rightmost “1” represents nviDigitalEnc_1, and the leftmost “1” represents “nviDigitalEnc_4.” Based on the combination of these inputs an alarm is triggered and a page is dialed.
- In this scenario two “Enable Trigger” boxes – corresponding to the logic described above - are checked. Sensors A & B are represented by “0 0 1 1” and sensors C & D are represented by “1 1 0 0”. The numbers following the trigger represent the event code portion of the page string. The “90011” and “91100” are the two event codes that would be sent based on which alarm condition occurs. This string may be set to any value that makes sense for your application.

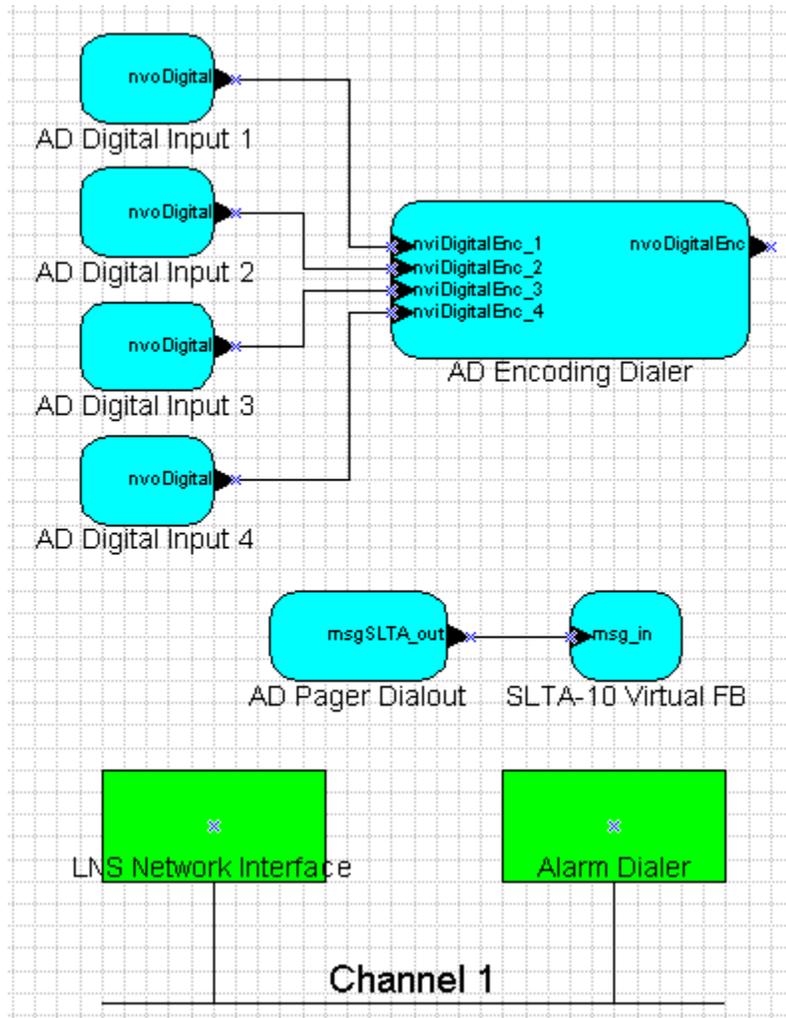
12. Click the “Dialing Configuration” tab.



- If an outside line is reached by first dialing a prefix, select the prefix checkbox and enter the prefix in the space provided. (This is useful, for example, if you have to first dial “9” to get an outside line.)
- Type the telephone number of the paging service in the “Pager number to dial” box.
- The “Waiting for pager beep delay” is the time in seconds the SLTA-10 will wait for the paging system prompt before sending off the page.
- A paging prefix may be used in front of the “Event code” to differentiate between different systems that all call the same pager. (This is generally set to the building number where the system is located)
- Check only the “Include I/O event in code page” since DM20/21-based alarms are not desired in this scenario.
- If the paging system requires a suffix at the end of a page, check the “Use paging suffix” box, and enter the suffix in the space provided. (For example, some pagers require you to hit the “#” key when you are finished keying in numbers.)
- The pager will receive numeric digits in the following form:



13. The complete LonMaker drawing should look something like this:



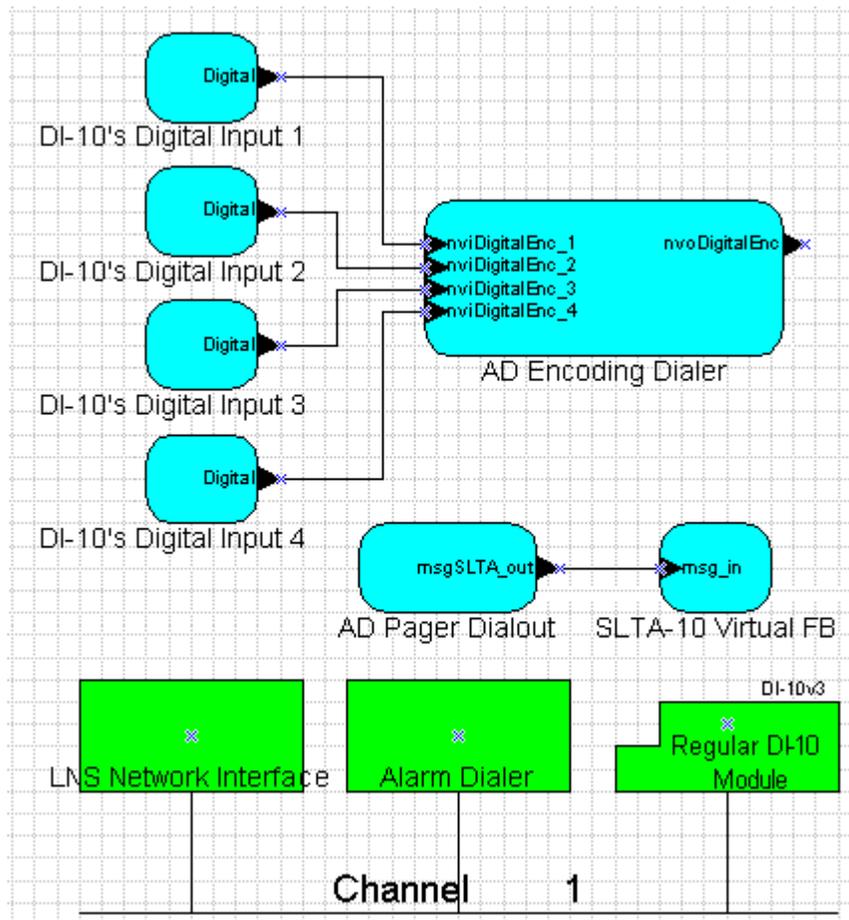
14. Save the LonMaker drawing

15. Remove the SLTA-10 RS-232 connection from your PC, and connect the cable to the modem. Be sure the modem is powered and connected to the telephone line. To test your system, set the input conditions on the Alarm Dialer node such that a page should be sent. When the alarm dialer detects the condition you should hear the modem take the telephone line off hook, and hear the DTMF tones required to call the paging service. Once the paging service answers, you should hear more DTMF tones to send the page. The modem should then hang up, and you should receive your page.

Scenario #2: Remote Hardware Fault

The Alarm Dialer's inputs may come from sensors located anywhere on the network. This could be another LonPoint DI-10 module, or any other device that supports LonMark "open loop sensor" objects with a SNVT_switch output. A regular DI-10 module will be used for this example.

Working with the same alarm conditions as example #1 above the only change to the LonMaker drawing is to connect the sensor inputs from the remote node to the Alarm Dialer instead of connecting the sensor inputs from the local node. The resulting LonMaker drawing should look something like:



Use the same procedures as outlined in example #1 above to set the trigger conditions and pager telephone number.

DM20/21 Event-based Alarm

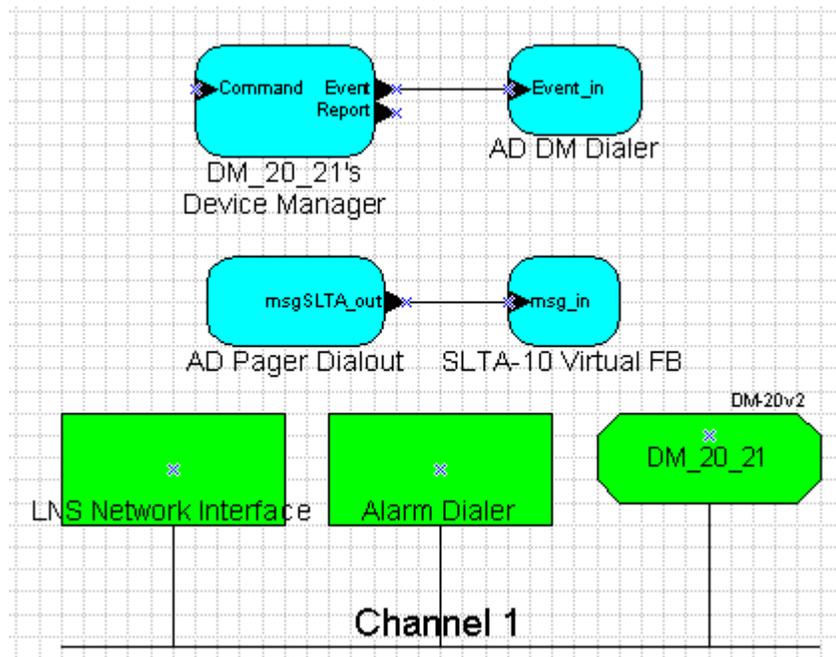
Alarms can also be triggered based on DM20/21 events. The DM-20/21 is an embedded network management device that remains on-site. It installs and monitors all the nodes in your network. The Alarm Dialer can trigger alarms (thereby sending a page) based on the following DM20/21 events.

- Database selected
- Event Log's limit reached
- Invalid database ID
- Variable location set
- Database selection overridden
- Fatal DM error
- DM20/21 Reset
- Unconfigured device found
- DM20/21 Reboot
- Configured device found
- Device Commissioned
- Device configuration error
- Device Commission Failed
- Device Missing

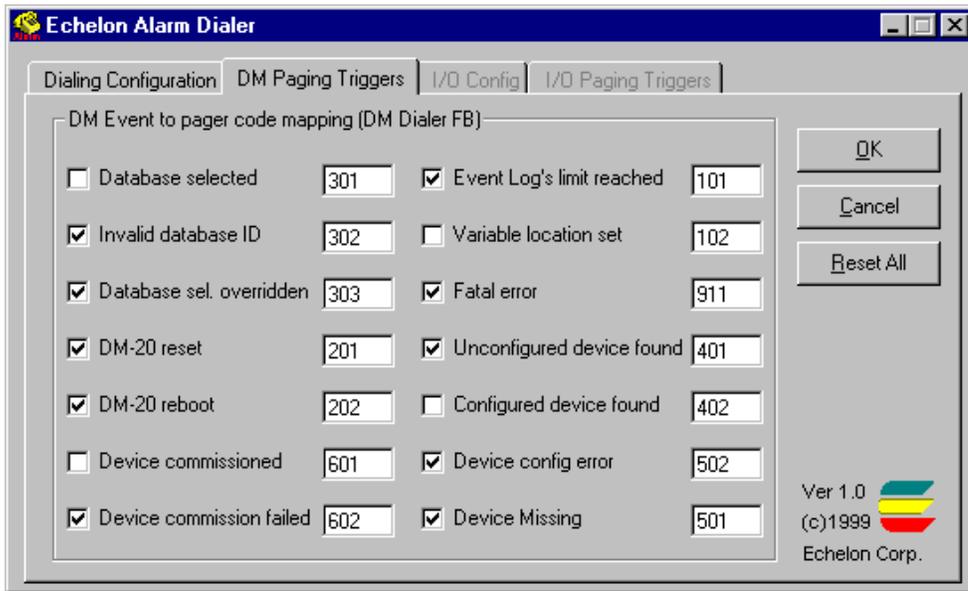
For this example, use an SLTA-10 as your network interface, but instead of directly installing the network use the DM loading wizard to compress the LonMaker drawing and load it into the DM-20. Be sure to use the SLTA-10 when you run the loading wizard to load the DM.

When the DM is loaded, disconnect the SLTA-10's RS-232 cable from the PC and connect it to the modem. Place the DM on the network and allow it to install the network as normal. (See the DM-20/21 users guide for complete details on using the DM-20/21)

1. Create the drawing diagrammed below using LonMaker for windows. The DM-20/21 shapes are installed as part of the DM-20/21 software installation. Use the DM-20/21 device stencil.



2. You must use the Alarm Dialer's plug-in to configure which DM events trigger a page. Note that events must be enabled on the DM-20/21 for this to work. See the DM-20/21 user's guide for complete information on DM-20/21 events.
3. Left-click on Alarm Dialer's "AD DM Dialer" functional block and select "Configure." The Alarm Dialer plug-in will pop up with the "DM Paging Triggers" tab selected. All DM related settings are configured via this tab. Select the checkbox next to each DM20/21 event that you want to cause a page. The numbers following the trigger checkbox represent the event code portion of the page string.



3

Functional Block & Plug-In Reference

This section explains how to use the Alarm Dialer Plug-in to configure the alarm processing software

Functional Blocks Overview

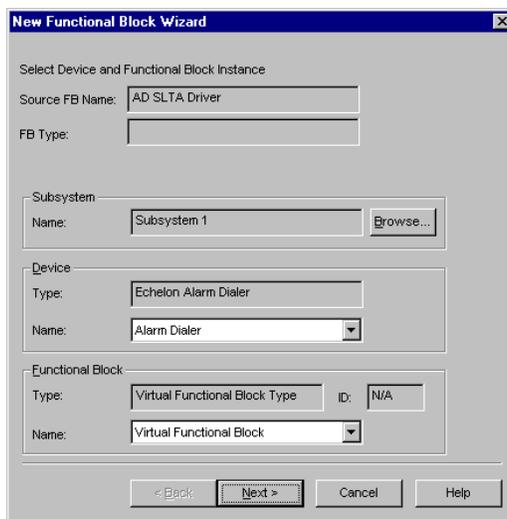
A functional block (or FB) is the logical representation for some specific software functionality residing on a physical device. Network variables and configuration properties reside in functional blocks. The Alarm Dialer contains the following functional blocks.

- Node object
- Digital Input[0]
- Digital Input[1]
- Digital Input[2]
- Digital Input[3]
- DM Dialer
- Encoding Dialer
- Virtual Functional Block

Each functional block represents a configurable software feature. See the LonMaker for Windows Users Guide for complete information on functional blocks and how to use them in LonMaker for windows.

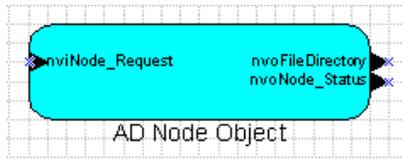
Virtual Functional Block

Application Messages are used to instruct the SLTA-10 to dial the modem. The Virtual functional block is the only functional block that can contain a message tag. The Alarm Dialer uses the msgSLTA_out tag to communicate to the SLTA-10.



Node Object

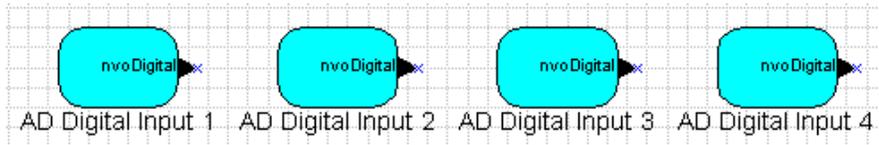
Standard LonMark node object



Name	Dir	Type	Definition
nviNode_Request	Input	SNVT_obj_request	Implemented for LonMark compliance of "Override" feature. Do not bind.
nviNode_Status	Input	SNVT_obj_status	Implemented for LonMark compliance of "Override" feature. Do not bind.
nviFile_Directory	Input	SNVT_address	Implemented for support of direct memory read/write configuration properties. Do not bind.

Digital Inputs

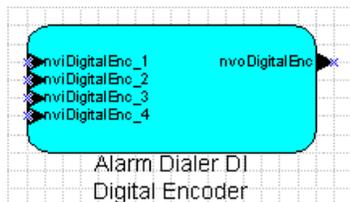
Open loop digital sensor objects similar to those found in Echelon's LonPoint DI module.



Name	Dir	Type	Definition
NvoDigital_1	Output	SNVT_switch	These output network variables represent the Alarm Dialer module's four physical inputs. Input behavior can be configured using the Alarm Dialer plug-in.
NvoDigital_2	Output	SNVT_switch	
NvoDigital_3	Output	SNVT_switch	
NvoDigital_4	Output	SNVT_switch	

Digital Encoder

Controller object similar to those found in Echelon's LonPoint DI module.



Name	Dir	Type	Definition
nviDigitalEnc_1	Input	SNVT_switch	Alarm conditions are triggered based on the values of these inputs. Each input combination can be mapped to a unique page. The Alarm Dialer plug-in is used to configure this object.
nviDigitalEnc_2	Input	SNVT_switch	
nviDigitalEnc_3	Input	SNVT_switch	
nviDigitalEnc_4	Input	SNVT_switch	
nvoDigitalEnc	Output	SNVT_switch	This variable is set to true (100 1) while an alarm condition is being processed. The "Enable digital encoder output" option must be enabled for this feature to work.

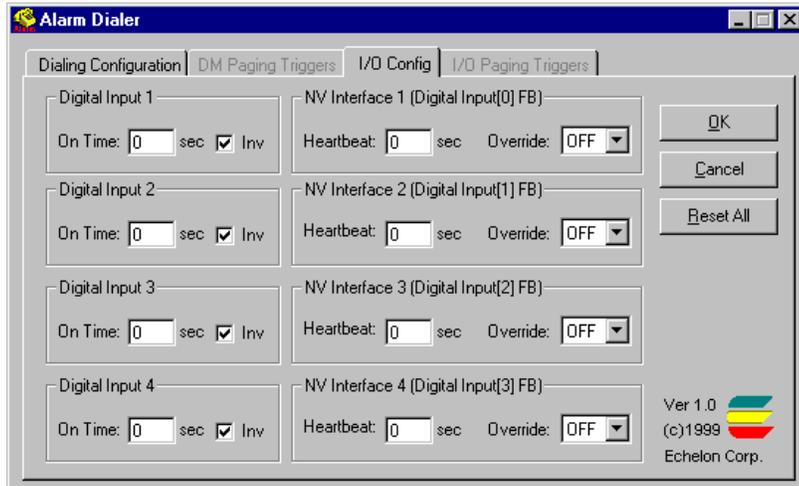
DM Dialer



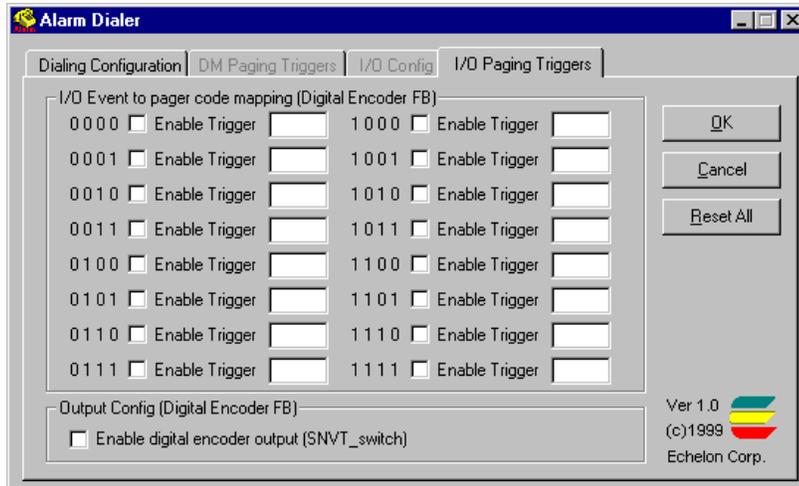
Name	Direction	Type	Definition
Event_in	Input	DmEvent	This functional block is used when triggering alarm conditions based on DM20/21 events. Bind the Event_in network variable to the DM20/21's "DM_Event" output network variable.

Plug-In Reference

The behavior of the Digital Input, Encoder, and DM Dialer functional blocks is configured using the Alarm Dialer plug-in. The plug-in is organized into three tabs, one for each of the functional blocks.



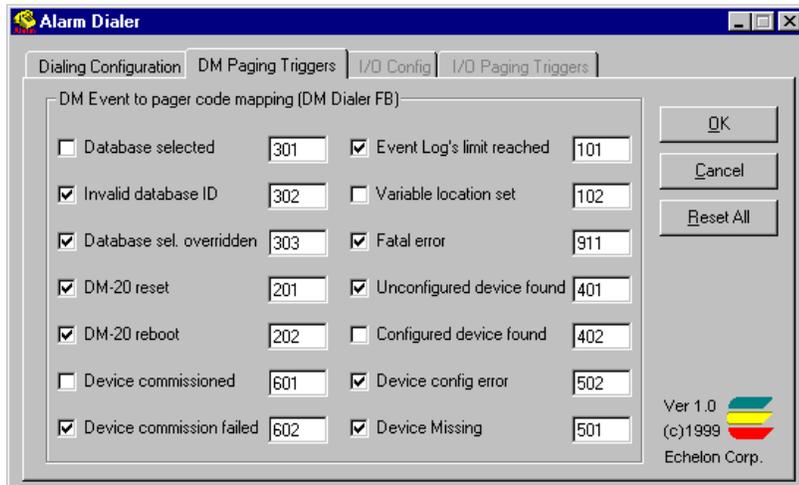
Frame	Option	Valid Range	Definition
Digital Input 1	On Time	0 – 64 sec	The physical input must remain 100ms + ‘On Time’ sec in the logically high state before the software will recognize the input in that position.
Digital Input 2	On Time	0 – 64 sec	
Digital Input 3	On Time	0 – 64 sec	
Digital Input 4	On Time	0 – 64 sec	
Digital Input 1	Inv	YES/NO	For an active high system, where 1 means ON, leave unchecked. For an active low system, where 1 means OFF, check this option.
Digital Input 2	Inv	YES/NO	
Digital Input 3	Inv	YES/NO	
Digital Input 4	Inv	YES/NO	
NV Interface 1	Heartbeat	0-9999 sec	The network variables nvoDigital_1 - nvoDigital_4 are updated every ‘heartbeat’ number of seconds. If the state of a physical input on the DI changes, the corresponding NV will also get updated. Set to 0 if no heartbeat updates are required.
NV Interface 2	Heartbeat	0-9999 sec	
NV Interface 3	Heartbeat	0-9999 sec	
NV Interface 4	Heartbeat	0-9999 sec	
NV Interface 1	Override	ON/OFF	If a Digital Input function block is in override mode, the DI’s physical input states will be ignored, and the network variable will be updated with the “override” value. To set an FB’s override status, left-click on the FB and select “Manage”
NV Interface 2	Override	ON/OFF	
NV Interface 3	Override	ON/OFF	
NV Interface 4	Override	ON/OFF	



The *IO event to pager code mapping* frame defines which bit combinations trigger pages. Bits are represented with the least significant bit as the right most bit. Digital input 0 maps to the least significant bit and digital input 3 maps to the most significant bit.

Enable the checkbox for each bit combination that you want to cause a page. Optionally, you may enter an event code in the text box provided next to each bit combination. This event code will be sent as part of the numeric page.

The Digital Encoder can also be used to feed other functional blocks much like the digital encoders available in Echelon's DI and DO LonPoint modules. To enable the encoder's output NV, select the checkbox in the *Output Config* frame.



The *DM paging triggers* tab is used to configure which DM events cause pages to be sent. Select the checkbox next to each DM event you want to send a page. Remember that the event must be enabled in the DM-20/21. See the DM-20/21 user's guide for complete information on enabling/disabling DM events.

Like other event triggers, you may assign a unique event code for each DM event. This event code is sent as part of the page string.

Appendix A

Plug-in Response Messages

This appendix documents the error and informational messages that may be issued by the Alarm Dialer Plug-in.

Informational Messages

Message	The device is not commissioned: all configuration data will be written to the LNS database only
Reason	Regardless of the network mode LonMaker opened the network with (OnNet/OffNet), the device is not properly commissioned. Therefore, the plug-in will only write the configuration property values to the LNS database only.
Solution	Commission the device with the default “Current Values in Database” option. During this process, the updated values for the configuration properties will be downloaded into the device.

Message	Operating in 'Off-Net' mode: all configuration data will be written to the LNS database only
Reason	The network was opened with LonMaker in the OffNet mode. All configuration properties (the configuration data specified via the plug-in) will be written to the LNS database only. The message will appear even if the device is properly commissioned.
Solution	Restart LonMaker and open the network in OnNet mode. Make sure the device is successfully commissioned, and then run the plug-in again. Clicking on the “OK” button in the plug-in will attempt to write the values in the LNS database to the device.

Error Messages

Message	A page must consist of: one or more ([pagingPrefix] [pagerDM_Event] [pagerIO_Event]) and optionally [pagingSuffix].
Reason	The Alarm Dialer must have some numerical information to pass to the pager, once it dials the pager number. To guarantee this, either the “Using Paging Prefix,” “Include DM Event Code in Page”, or the “Include I/O Event Code in Page” option in the “Dialing Configuration” tab must be set. A paging suffix does not constitute a page.
Solution	Select one of the aforementioned checkboxes and this error will not appear.

Message	The 'XXX' string cannot be longer than 'XXX' characters. A maximum of 'XXX' digits can represent the 'XXX'. A maximum of 'XXX' can represent the 'XXX'.
Reason	These errors appear when the user attempts to enter more data for a text string, than there is space allocated for it on the node. It is important to note that all parameters configured with the plug-in will eventually reside as data on the node.
Solution	Select a shorter text string size. The message will be specific enough to determine which string is longer than expected.

Message	The 'XXX' must be from 'XXXX' to 'XXXX' seconds long. A 'XXX' must be a numerical value.
Reason	These errors appear when the user attempts to enter a numeric value outside the range of possible values for the configuration parameter.
Solution	Select a parameter value within the proper range. The message will be specific enough to determine which parameter is out of range, and what the allowed range is.

Appendix B

Data Structures

This appendix documents the data structures used to store the Alarm Dialer configuration properties in the LNS database and on the Alarm Dialer module.

Alarm Dialer CP Structures

```
typedef struct
{
    char str[25];
} UCPT_DMD_ModemInitString;

typedef struct
{
    char str[25];
} UCPT_DMD_ModemDialString;

typedef struct
{
    char str[25];
} UCPT_DMD_ModemHangupString;

typedef struct
{
    short fUseOutsideLinePrefix;
    char outsideLinePrefix[6];
    char pagerNumber[24];
} UCPT_DMD_CallingOptions;

typedef struct
{
    short pagerBeepDelay;
    short fUsePagingPrefix;
    char pagingPrefix[6];
    short fIncludeDMEvents;
    short fIncludeIOEvents;
    short fIncludeNVEvents;
    short fUsePagingSuffix;
    char pagingSuffix[6];
} UCPT_DMD_PagingOptions;

typedef struct
{
    unsigned short fEventLogLimit;
    unsigned short fDMReset;
    unsigned short fDMReboot;
    unsigned short fDatabaseSelected;
    unsigned short fInvalidDatabaseID;
    unsigned short fDBSelectOverridden;
    unsigned short fFoundUncnfg;
    unsigned short fFoundCnfg;
    unsigned short fDeviceMissing;
    unsigned short fDeviceConfigError;
    unsigned short fDeviceCommissioned;
    unsigned short fDeviceCommissionFailed;
    unsigned short fFatalError;
    unsigned short fVariableLocationSet;
} UCPT_DMD_PagingTriggers;

typedef struct
```

```

{
    unsigned char cdeEventLogLimit[6];
    unsigned char cdeDMReset[6];
    unsigned char cdeDMReboot[6];
    unsigned char cdeDatabaseSelected[6];
    unsigned char cdeInvalidDatabaseID[6];
    unsigned char cdeDBSelectOverridden[6];
    unsigned char cdeFoundUncnfg[6];
    unsigned char cdeFoundCnfg[6];
    unsigned char cdeDeviceMissing[6];
    unsigned char cdeDeviceConfigError[6];
    unsigned char cdeDeviceCommissioned[6];
    unsigned char cdeDeviceCommissionFailed[6];
    unsigned char cdeFatalError[6];
    unsigned char cdeVariableLocationSet[6];
} UCPT_DMD_PagerEventCodes;

typedef struct
{
    char debounce[6];
    char fInvert;
} UCPT_DMD_DigitalInput_t;

typedef struct
{
    char heartbeat[6];
    char fOverrideVal;
}
UCPT_DMD_NVOutput_t;

typedef struct
{
    UCPT_DMD_DigitalInput_t  DI[4];
    UCPT_DMD_NVOutput_t     NV[4];
}
UCPT_DMD_IOConfig;

typedef struct
{
    char fEnabled;
    char pagerCode[6];
} UCPT_DMD_OneIOTrigger_t;

typedef struct
{
    UCPT_DMD_OneIOTrigger_t trigger[16];
    char fDigitalEncoder;
} UCPT_DMD_IOPagingTriggers;

```