Technical Note

Setting Up the EC9537 SMS7000 Service

This note describes the EC9537 SMS Serial Node Controller Service.

The service is software that allows serial communication between GVG SMS7000 routers and NV9000 router control systems using either standard or enhanced node controllers, and matrix controllers in the SMS 7500. It also supports an expanded 256×256 router with enhanced node controllers.

The service runs on the active NV9000 system controller (and on the stand-by system controller, if you have a redundant system). It executes independently of the NV9000 software, but communicates with, and is controlled by, the NV9000 software. (Starting and stopping the NV9000 software has no effect on the service.)

The service software is typically shipped on a CD (SB0227) and is identified by any of 3 part numbers: EC9537, SP0105, or SV0929. The part numbers all refer to the same product.

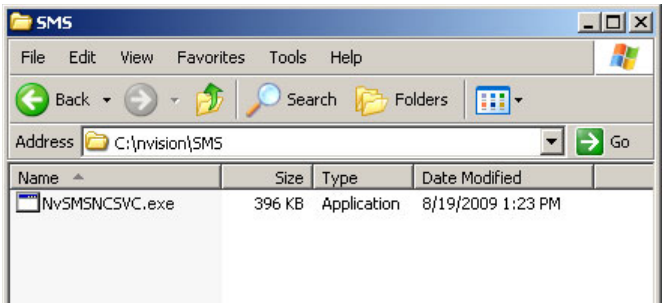
Installation

Follow these steps:

1 Connect a keyboard, monitor, and mouse to your NV9000 system controller if you have not done so previously. Log in. The default user ID and password are ‘envyadmin’ and ‘software’, respectively.

2 Copy the software from the CD to the NV9000.

The software is the file named NvSMSNCSVC.exe. You must copy it to the folder C:\nvision\sms.

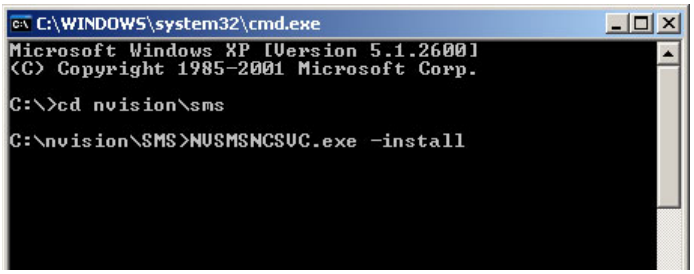


Create the folder if it does not exist.

3 Install the service.

Open the command prompt window. Go to the \nvision\sms directory. At the prompt, type

```
NVSMSNCSVC.exe -install
```



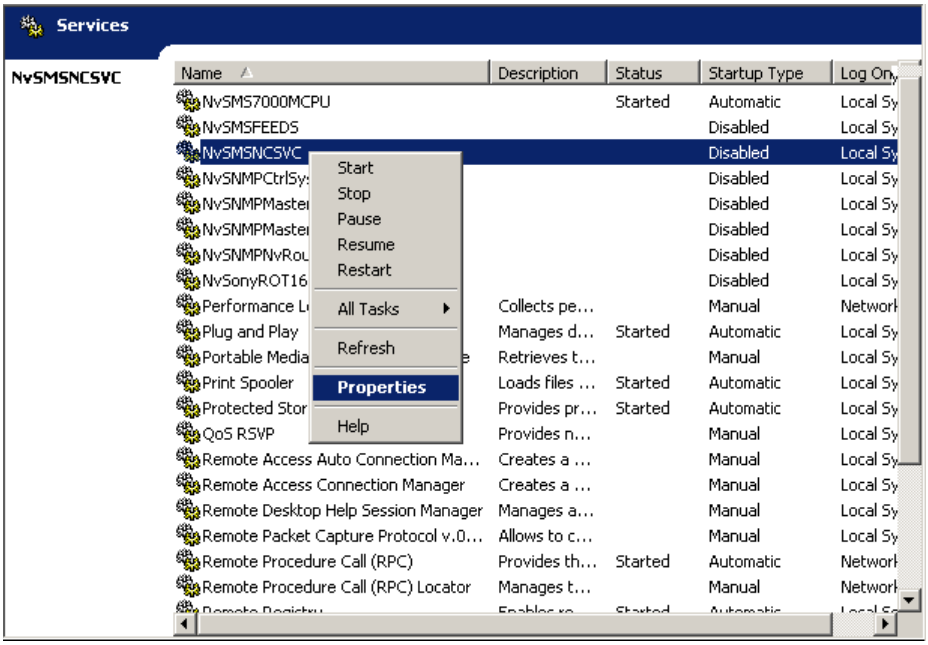
To uninstall the software, you would type `NVSMSNCSVC.exe -uninstall`.

4 Ensure that the service restarts automatically when the NV9000 system controller restarts.

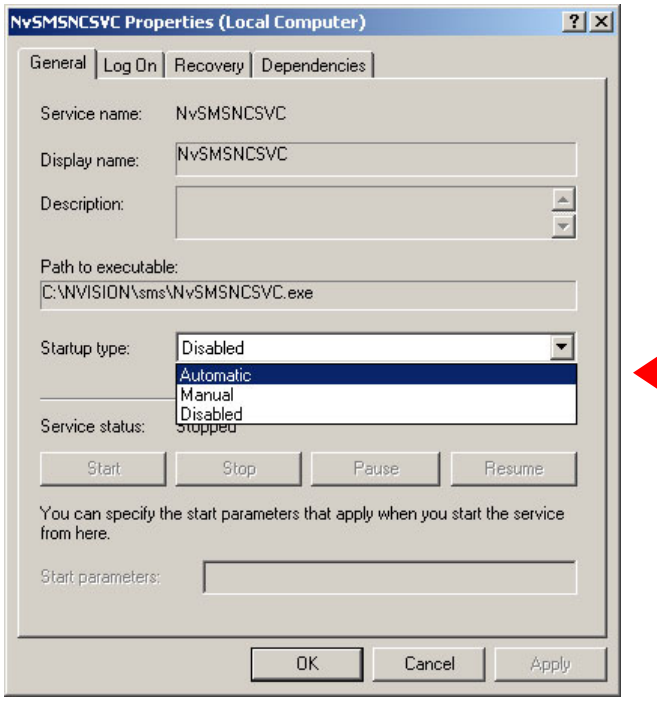
Right-click the system controller's icon on the Windows desktop of the NV9000 system controller. (Typically the name of the system controller will be NVCONFIG or some variation of NVCONFIG.) Choose 'Manage' from the context menu that appears.

Then choose 'Services and Applications' on the left and from that choose 'Services'.

Locate the software in the services list. Right-click the software and choose 'Properties' from the context menu:



A tabbed window then appears. Choose 'Automatic' for the startup type and click OK:



You can also get to the services window through the Windows control panel. Choose ‘Administrative Tools’ and then ‘Services’.

5 Repeat the preceding steps for the secondary system controller if you have a redundant NV9000 system.

Initial Configuration

There are two parts to this configuration step:

- Define a configuration file for each SMS router the NV9000 is to control.
- Identify those routers in NV9000-SE Utilities.

Ports

A “port” is an internal, not external, Ethernet port through which the NV9000 software will communicate with the SMS service. The ports are numbered. For every configuration file — for every SMS router you want to control — you will choose a port number.

▲ The NV9000 software communicates with the port as if it were the actual router. (The NV9000 controller is connected to the actual SMS router(s) using one or more serial cables.)

Choose port numbers in one of these ranges for “well-known ports”:

9451–9499
9630–9699
9701–9746
9803–9874

The numbers you choose will be used for the configuration files and for configuration entries in NV9000-SE Utilities.

If you have a stand-by system controller, the SMS service will automatically create additional and identical ports for communcating with the stand-by system controller. The number of a secondary port is 10000 + the number you gave to a primary port.

Configuration Files

A port configuration file is a plain text file with a `.cfg` extension. The port configuration file must reside in the C:\nvision\sms folder mentioned under Installation.

The name of the configuration file must be the concatenation of “SMS” and the port number. For example, if the port is 9457, the configuration file name is SMS9457.cfg.

There are two types of configuration files: standard and extended. Examples of each type are given in the Appendix on page 14. The standard configuration file supports standard, or non-expanded frames only. The extended configuration file supports either standard or expanded frames.

We recommend that you create a copy of the port configuration files given and modify their paramers to suit your system.

You must place these files on both system controllers if you have a redundant NV9000 system.

NV9000-SE Utilities

There are two places to make configuration entries for the “SMS routers”:

- Control Points table.

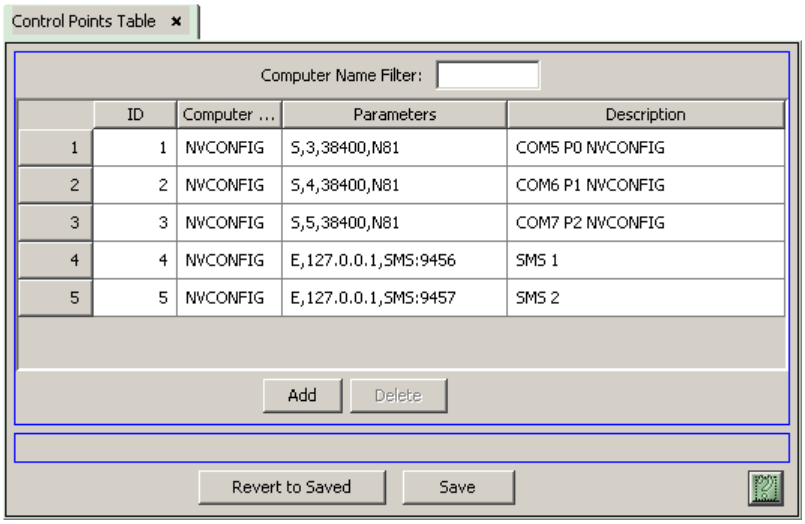
- Routers list.

It is important to make the configuration entries (and deletions) in the order expressed here.

Add Control Points

Go to the ‘Control Points’ table under the ‘Views’ navigation pane. Add a control point for each “SMS router” your NV9000 controls. For example, if the port you have chosen is 9456, enter

```
E,127.0.0.1,SMS:9456
```

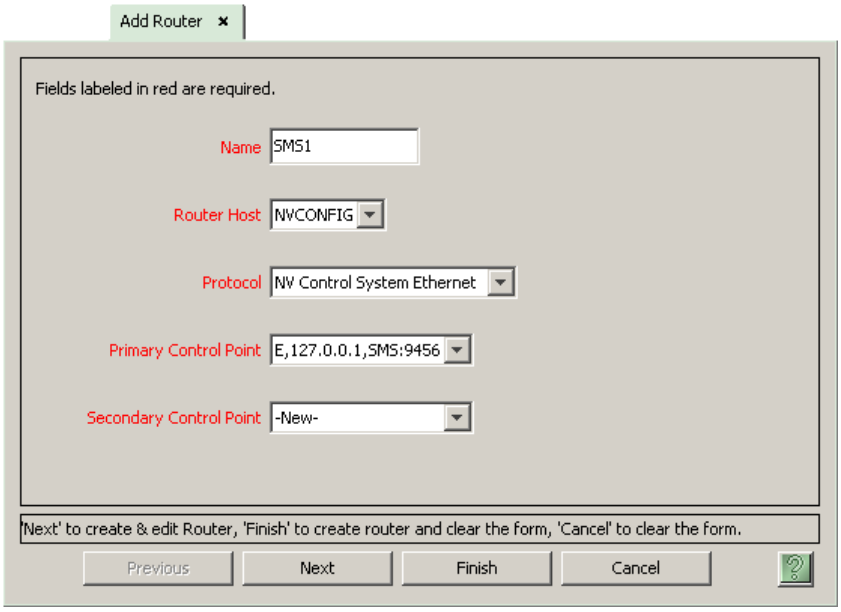


To make the entry, you must first click the ‘Add’ button. It is essential to click the ‘Save’ button to commit your change.

The IP address 127.0.0.1 is a special internal “loopback” address that can be used by the NV9000 for the SMS service. Use the same address in each entry, but qualify the addresses with the SMS port number.

Add Routers

Go to the ‘Routers’ page under the ‘Configuration’ navigation pane. Click ‘Add Router’. The following page appears:



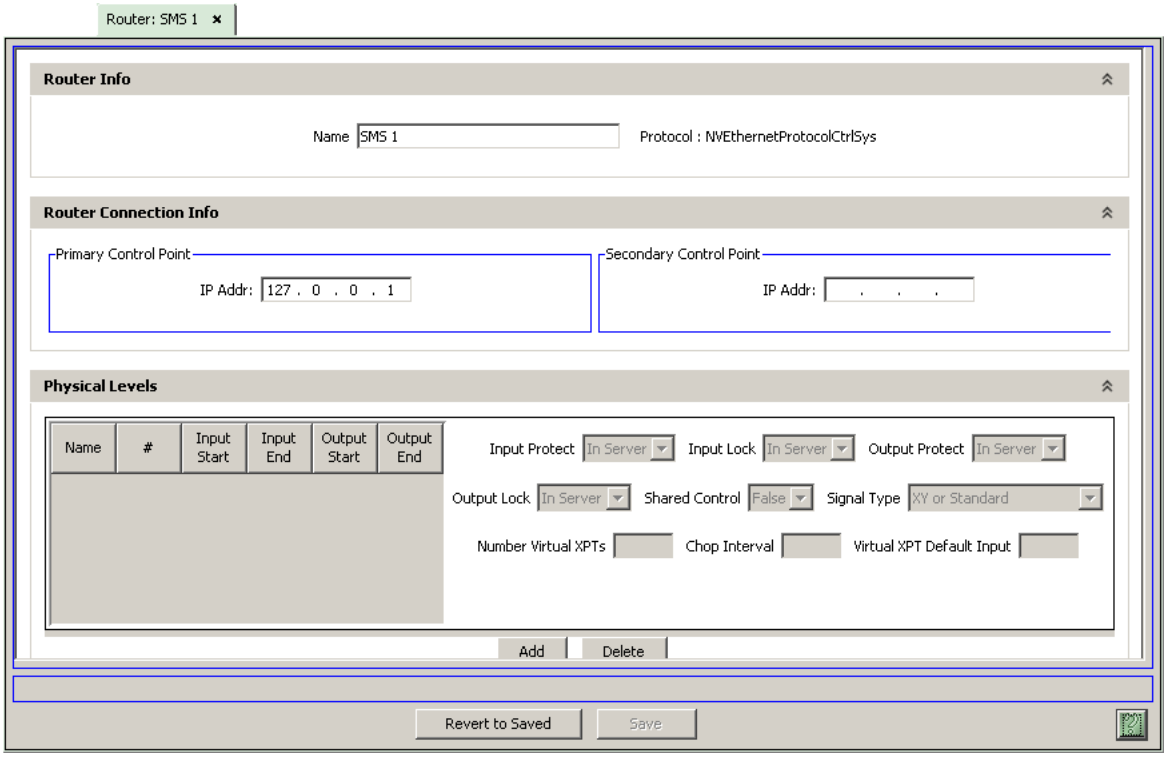
Enter values in all fields except ‘Secondary Control Point’.

The important fields are ‘Protocol’ and ‘Primary Control Point’. The protocol must be ‘Nv Control System Ethernet. The primary control point must be one of the control points you defined for an “SMS router.”

The name you give to the router should be meaningful.

Choose a router host. Generally, the router host is NVCONFIG.

Having entered values in the ‘Add Router’ page, click ‘Next’. The router details page appears:



It reflects what you entered on the ‘Add Routers’ page.

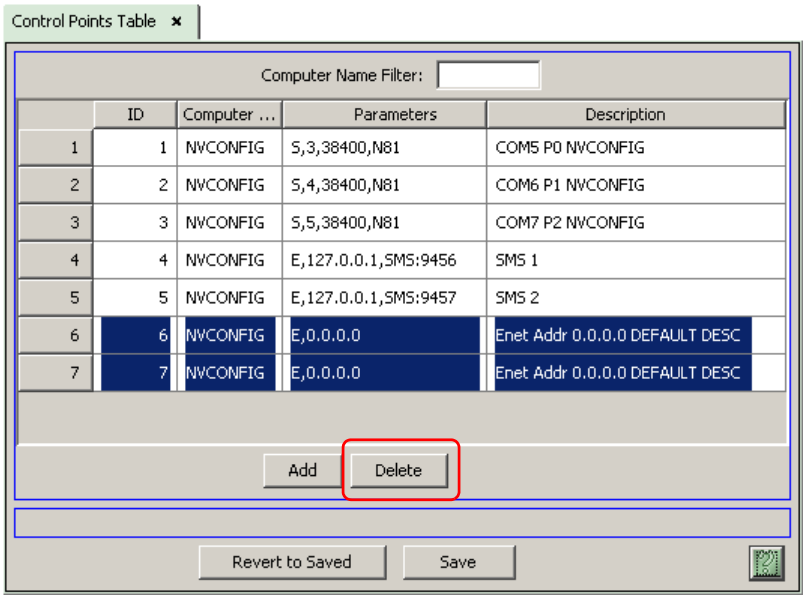
NV9000-SE Utilities will have created an IP address in the ‘Secondary Control Point’ field. Erase that IP address. That will change the IP address to all zeros, an easily identifiable address that you will need in the next step.

It is in this page that you will eventually add physical levels and configure the options for the physical levels. It is not necessary to do so at this point in your configuration.

Delete Extraneous Control Points

Return to the ‘Control Points ’ table under the ‘Views’ navigation pane. Highlight any extra control points that were created — as secondary control points — when you added the routers. Of course, highlight only those you don’t want. Click ‘Delete’ to delete the extra control points.

This example shows two extra control points that are identifiable because their IP addresses are 0.0.0.0:



Summary

For each SMS router you are controlling, you will have created a configuration file (.cfg). The name of the file identifies an internal NV9000 port. The SMS service software communicates with SMS router through its assigned port.

For each SMS router, there will be an entry in the control points table, identifying the port assigned.

For each SMS router, there will be an entry in the routers table.

Connections

We present four connection scenarios, each involving a redundant NV9000 system, that is, a primary system control and and secondary system controller. (One is active; the other is stand-by. In a fail-over, the stand-by controller becomes active.)

Figures 1–4, following, show the various ways of connecting NV9000 system controllers to SMS frames. The connections use one or more WC0137 cables, Moxa cards and breakout boxes, and possibly additional Miranda equipment (such as NV9500).

Connection details follow the scenarios.

Scenario 1

This scenario is a connection to a standard SMS router with standard node controllers:

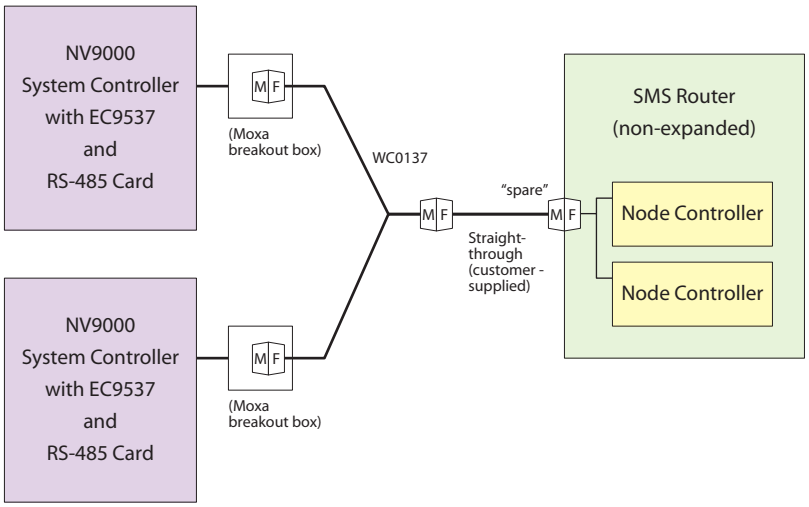


Figure 1. Standard Frame, Standard Node Controller(s)

The scenario requires one WC0137 "Y" cable.

Scenario 2

This scenario is a connection to an expanded SMS router with standard node controllers:

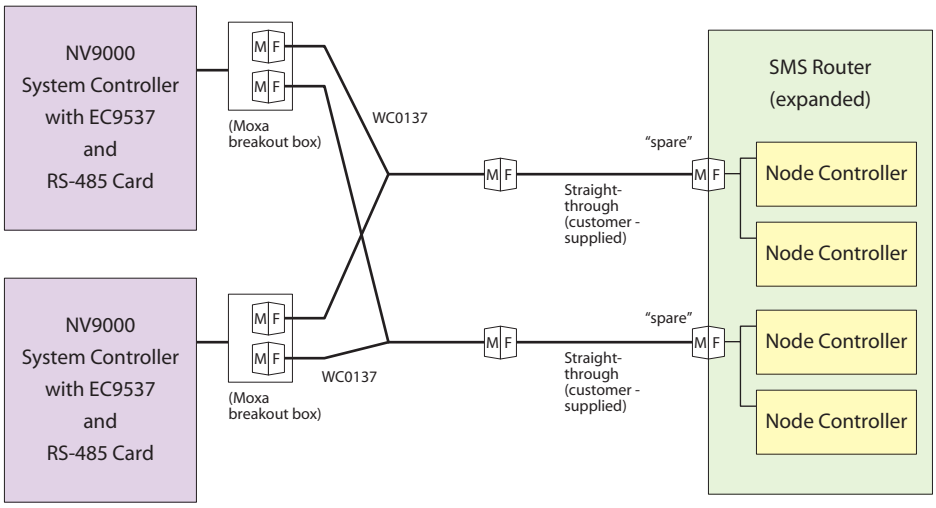


Figure 2. Expanded Frame, Standard Node Controller(s)

The scenario requires two WC0137 "Y" cables.

Scenario 3

This scenario is a connection to an standard SMS router with enhanced node controllers:

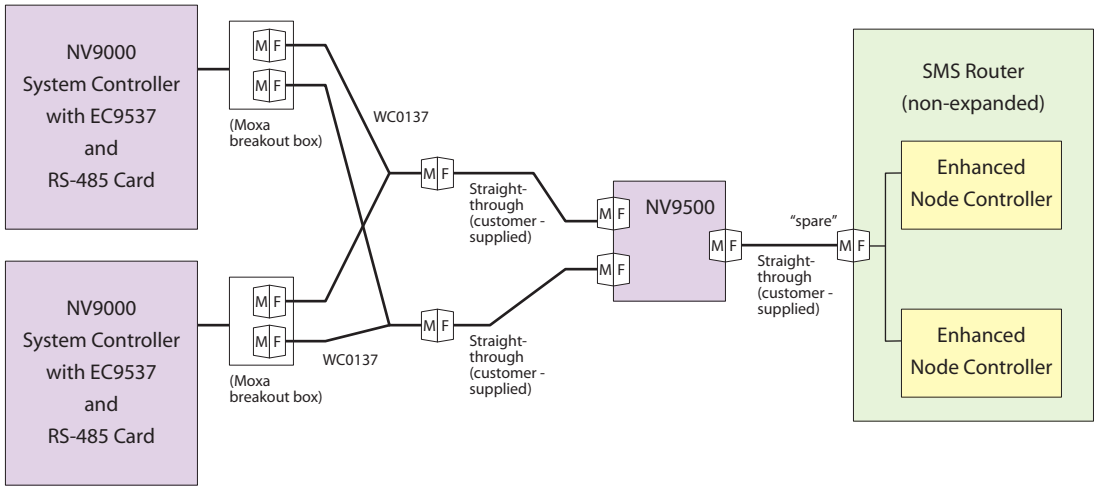


Figure 3. Standard Frame, Enhanced Node Controller(s)

This scenario requires two WC0137 "Y" cables and one NV9500 (enhanced node controller interface).

Scenario 4

This scenario is a connection to an expanded SMS router with enhanced node controllers:

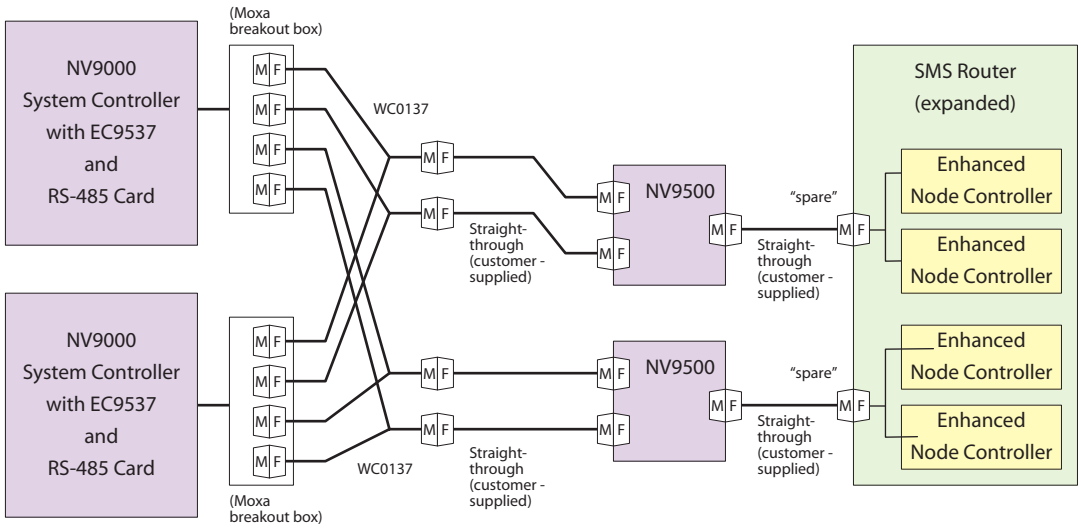


Figure 4. Expanded Frame, Enhanced Node Controller(s)

This scenario requires four WC0137 "Y" cables and two NV9500s (enhanced node controller interfaces).

Serial Expansion Ports

The NV9000 requires a serial expansion card (part number EC9420) in one of its expansion slots. The EC9420 provides 8 serial channels and comes with an 8-port breakout box. It is to the breakout box (on either the primary or secondary controllers) that you will make cable connections.

Serial Cables

WC0137 Cable

The WC0137 cable connects the Moxa breakout box of the NV9000 to either an SMS router's "spare" serial port or to an RS-422/485 port of the NV9500 interface. The Moxa pinout differs from the SMS pinout. The WC0137 cable accounts for the difference:

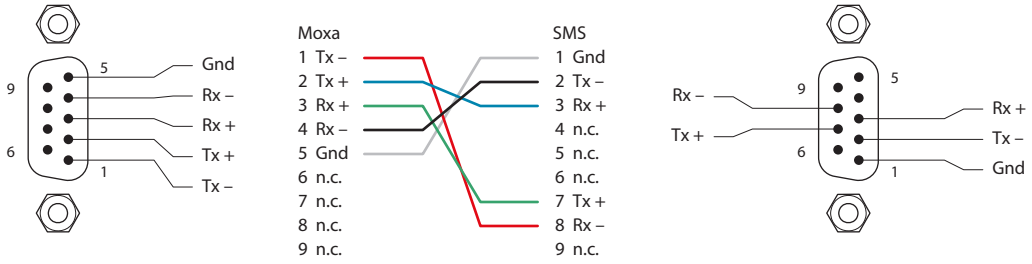


Figure 5. WC0137 Pinouts

Straight-through cabling can used on either end of the WC0137 ("Y") cable if you need additional length.

The SMS pinout shown here is not applicable to enhanced node controllers. It is the NV9500 that connects to enhanced node controllers.

NV9500

The NV9500 converts the RS-485 signals from the NV9000 System Controller to RS-232 for the SMS7000 enhanced node controller. It has two converters, P1 and P2, one for each system controller in a redundant NV9000 system.

To keep costs to a minimum, the NV9500 uses an open chassis frame. It is a "shelf."

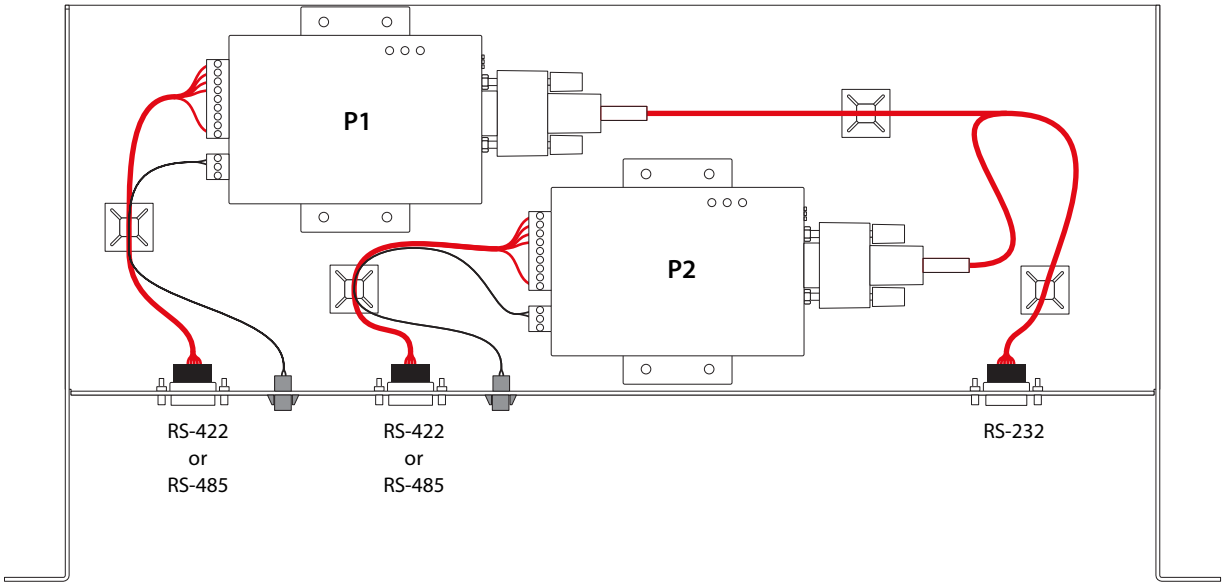
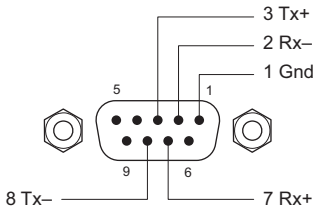


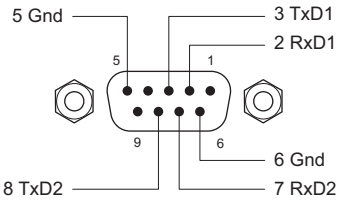
Figure 6. NV9500, Top View

Figure 7. NV9500, Front View

Figure 8 shows the pin assignments of the three DE9 connectors on the NV9500:



RS-422/485 Converters 1 and 2



RS-232 to SMS Router

Figure 8. Serial Ports

A WC0137 “Y” cable connects one of the NV9000’s Moxa breakout boxes to the NV9500 (at one of the RS-422/RS-485 ports). Two “Y” cables are required for each NV9500 in a redundant NV9000 system. Two NV9500s are required for an expanded SMS router. The RS-232 port of the NV9500 connects to a “spare” port of the SMS router, as depicted in figures 3 and 4.

Use straight-through cables to extend the Y cables if necessary.

Maintenance and Changes

Restart

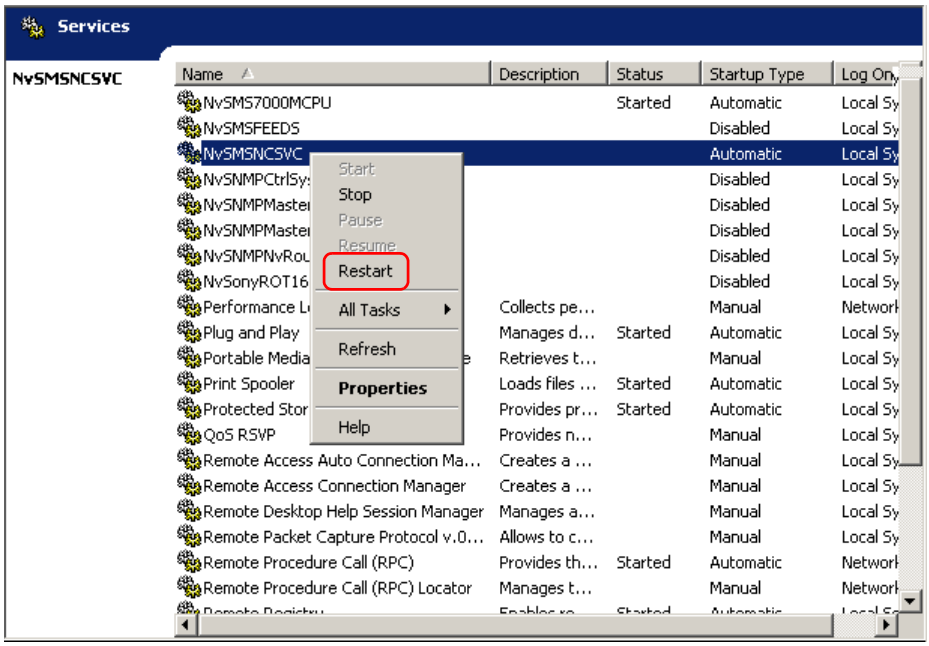
If you make changes to the SMS service (in NV9000-SE Utilities), you must restart both the service and the NV9000 software after writing the configuration changes to the NV9000.

Follow these steps to restart the SMS service:

1 Right-click the system controller’s icon on the Windows desktop of the NV9000 system controller. Choose ‘Manage’ from the context menu that appears.

2 Choose ‘Services and Applications’ on the left and from that choose ‘Services’.

3 Right-click NvSMSNCSVC in the services list. Choose ‘Restart’ from the context menu:



Version Number

The software version number (SV0929-xx) is entered in the NV9000 log at the start of logging. You can check the log file if you are interested. Technical support personnel might ask for this version number. (The log file is

```
C:\nvision\envy\userlocal\logs\NVSMSNCSVC.log
```

or it might be

```
D:\nvision\envy\userlocal\logs\NVSMSNCSVC.log.)
```

You can also use telnet. To do so, open the Windows command prompt window and type ‘telnet localhost 9168’ at the prompt. A DOS-type telnet window opens and you can see log entries in real-time. You can also type a digit from 1 to 6 to set the logging level.

As of NV9000-SE Utilities version 5.2.2, you can view the logs from NV9000-SE Utilities. Select ‘SMS NC SVC log’ from the log viewer’s drop-down list.

SMS Node Controller Modifications

You might need to perform some board rework of the SMS node controllers to support redundant operation under NV9000 serial control.

Overview

Because the serial communication to the SMS node controllers is RS-485, on shared wiring, with no addressing mechanism available in the control protocol, both node controllers receive the same commands at the same time. Both become active at the same time. Only one must become active; the other must remain stand-by.

A hardware modification makes one node controller card dominant. The dominant card will always be the active card unless it goes unhealthy or is removed, at which point the non-dominant card goes active.

Circuit Detail

The input to each card has an R/C network to keep the cards from going active too soon after insertion into the backplane. (The original values are 100 kΩ for the series resistor and 1 µF for the capacitor.)

When both cards are like this, they both try to go active and as a result neither one is successful.

Solution

On the board meant to be dominant, a smaller resistor value was needed to pull the line down harder when the backplane signal asserts low. That required a capacitor change to maintain the original timing constant. The values selected were a 30.9 kΩ resistor and a 0.33 µF capacitor.

Vary the timing between the two cards to allow one to become fully active before the other. This allows the NV9000’s serial control protocol to communicate with only one controller.

Reworking the Node Controllers

Rework is very simple and inexpensive. The node controllers are modified in A/B pairs.

Node Controller “A” has resistor X5R38 changed to 56.2 kΩ and capacitor X5R41 changed to 4.7 µF.

Node Controller “B” has resistor X5R38 changed to 30.9 kΩ and capacitor X5R41 changed to 0.33 µF.

Values of 56.2 kΩ and 4.7 µF yield a 520 ms delay for the dominant card.

The components can be found on the board not far from the lithium battery against the backplane connector:

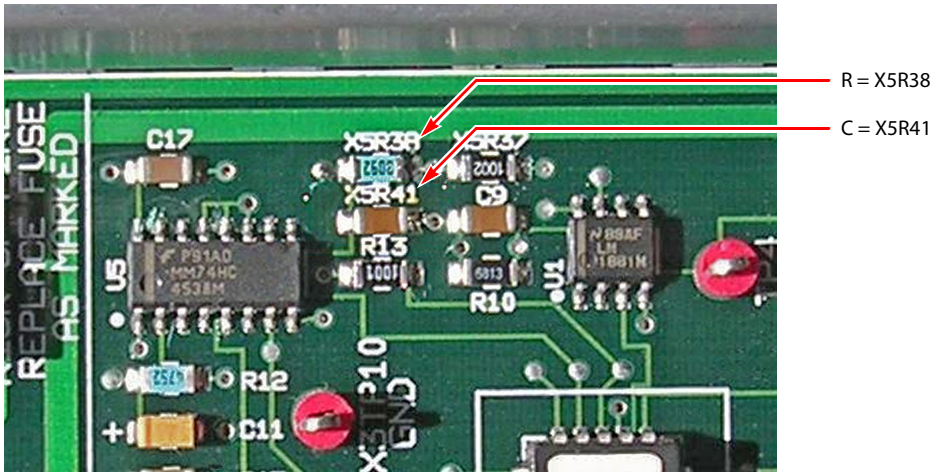


Figure 9. Capacitor X5R41 Location

We recommend you label the boards as A and B so the labels can be seen when the boards are installed in the frame. It is advisable to modify and label spare maintenance stock too.

Additional Configuration

It is necessary to perform NV9000 configuration in general to allow the NV9000 to control the SMS router crosspoints, and I/O devices connected to the SMS routers.

This note does not address that level of NV9000 configuration.

Appendix

Standard Configuration File

Following is a sample configuration for a standard SM router:

```
;**********************************
; Configured with standard node controllers in single frame
; Values shown are in decimal
;**********************************
;* MaxSlice can be (0-4) usually 0
MaxSlice = 0
;**********************************
; * Slice can be (0 - 4) usually 0. Nonzero is only for expanded
Slice=0
;**********************************
; * SignalType can be the following:  (these are the known - there may be others)
;   (these values have 0x20(32) added to them for exclusive external control)
; * 32 = ST_NTSC                (0x20 + 0x00) - 143Mbs
; * 33 = ST_D2_PAL              (0x20 + 0x01) - 177Mbs
; * 34 = ST_D1                  (0x20 + 0x02) - 270Mbs smpte259
; * 35 = ST_HDTV                (0x20 + 0x03) - compressed hd 360Mbs smpte259
; * 36 = ST_ANALOG_VIDEO        (0x20 + 0x04)
; * 37 = ST_ANALOG_VIDEO_DCR_ON (0x20 + 0x05)
; * 38 = ST_OUTPUT_FOLLOW_IN    (0x20 + 0x06) - actually implemented in NC?
; * 39 = ST_BYPASS_RECLOCK      (0x20 + 0x07) - passes dig vid w/o reclocking
; * 40 = ST_DIGITAL_AUDIO_1     (0x20 + 0x08)
; * 41 = ST_ANALOG_AUDIO_1      (0x20 + 0x09)
; * 42 = ST_DIG_VID             (0x20 + 0x0A) - 540Mbs
; * 43 = ST_HIGHDEFINITION      (0x20 + 0x0B) - 1.485Gbps smpte292
; * 44 = ST_AUTO_RECLOCK        (0x20 + 0x0C) - reclocks dig vid
; *
;******note: two other values may be added to the signal type obtained from mcpu
;****** 0x00(0)  - ST_SHARED_CONTROL
;****** 0x40(64) - ST_MCPU_EXCLUSIVE_CONTROL
;****** These values will need to be subtracted off of the signal type and the
;****** 0x20(32) value added on to get the correct control.
SignalType   = 41
;**********************************
; * Frame Type can be the following:  (these are the known- there may be others)
; * 0  = FT_QUAD_32X32_AUDIO    (0x00)
; * 1  = FT_DUAL_64X64_AUDIO    (0x01)
; * 2  = FT_DUAL_64X32_AUDIO    (0x02)
; * 3  = FT_DUAL_32X64_AUDIO    (0x03)
; * 16 = FT_128X128_VIDEO       (0x10) - used for expanded type
; * 17 = FT_128X96_UPPER_VIDEO  (0x11) - ??
; * 18 = FT_128X96_LOWER_VIDEO  (0x12) - ??
; * 19 = FT_128X64_VIDEO        (0x13)
; * 32 = FT_128X64_AUDIO        (0x20)
; * 34 = FT_AES_NX256           (0x22) - NB256x256, NB512x512
; * 48 = FT_64X64_VIDEO         (0x30) - classic video
; * 64 = FT_32X32_VIDEO         (0x40)
; * 98 = FT_7500WB_256x256      (0x62) - 7500 WB256x256
; * 99 = FT_7500WB_128x128      (0x63) - 7500 WB128x128; *
FrameType = 1
;**********************************
; ConfigOutputSize = the number of outputs in the router
ConfigOutputSize = 64
```

```
;**********************************
; ConfigInputSize = the number of inputs in the router
ConfigInputSize = 64
;**********************************
; This gives an offset - have not seen it used - maybe with expanded
ConfigOutputOffset = 0
;**********************************
; This gives an offset - have not seen it used - maybe with expanded
ConfigInputOffset = 0
;**********************************
; ExpansionType can be the following:
; * 0 = ET_SINGLE   - which means non-expanded
; * 1 = ET_MULTIPLE - which means there are several frames (expanded)
ExpansionType = 0
;**********************************
; Have only seen 0 here so far??
ExpansionSize = 0
;**********************************
; WhichOutputs will let the NC know which set of up to 128 outputs it is controlling
; Values can be the following:
; * 0   = WO_128X0 - which is the standard for non-expanded.
; * 128 = WO_128X1 - ONLY VALID FOR EXPANDED
; * 256 = WO_128X2 - ONLY VALID FOR EXPANDED
WhichOutputs = 0
;**********************************
; NCType lets it know if it is dealing with Standard or Enhanced Node Controllers
; Values can be the following:
; * 0 = Standard node controllers
; * 1 = Enhanced node controllers
NCType = 0
;**********************************
; This lets the software know which com port to use.
; NCComPortOne is for standard node controllers or the first enhanced
NCComPortOne = 5
;**********************************
; FOR ENHANCED ONLY!
; This lets the software know which com port to use for the second node controller
; If there is only one enhanced node controller - please set to 0.
NCComPortTwo = 0
;///end///
```

Extended Configuration File

Following is a sample configuration for an extended SM router:

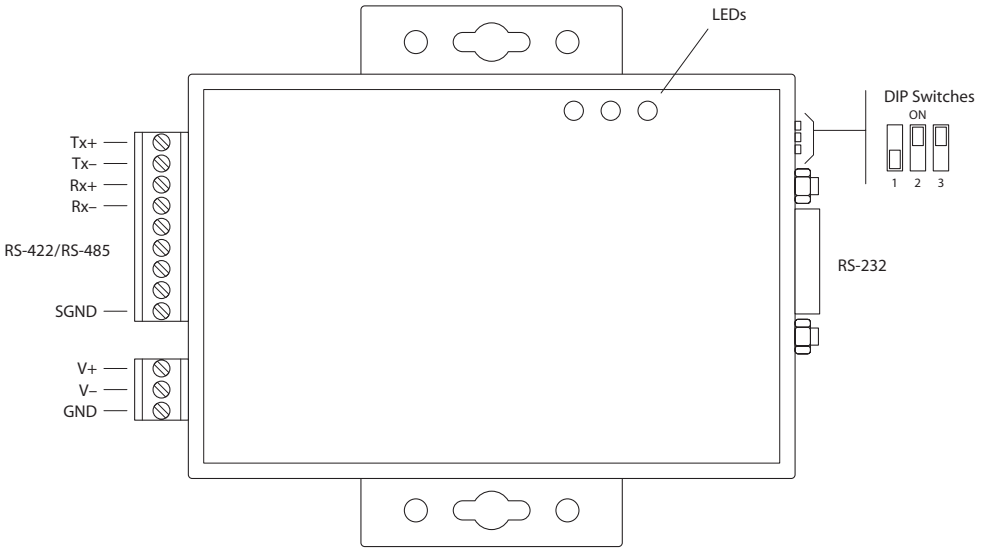
```
;**********************************
; Configured with enhanced node controllers in expanded frame
; Values shown are in decimal
;**********************************
; * MaxSlice can be (0-4) usually 0
MaxSlice = 0
;**********************************
; * Slice can be (0 - 4) usually 0. Nonzero is only for expanded
Slice=0
;**********************************
; * SignalType can be the following:
;   (these values have 0x20(32) added to them for exclusive external control)
; * 32 = ST_NTSC                 (0x20 + 0x00) - 143Mbs
; * 33 = ST_D2_PAL               (0x20 + 0x01) - 177Mbs
; * 34 = ST_D1                   (0x20 + 0x02) - 270Mbs smpte259
; * 35 = ST_HDTV                 (0x20 + 0x03) - compressed hd 360Mbs smpte259
; * 36 = ST_ANALOG_VIDEO         (0x20 + 0x04)
; * 37 = ST_ANALOG_VIDEO_DCR_ON  (0x20 + 0x05)
; * 38 = ST_OUTPUT_FOLLOW_IN     (0x20 + 0x06) - actually implemented in NC?
; * 39 = ST_BYPASS_RECLOCK       (0x20 + 0x07) - passes dig vid w/o reclocking
; * 40 = ST_DIGITAL_AUDIO_1      (0x20 + 0x08)
; * 41 = ST_ANALOG_AUDIO_1       (0x20 + 0x09)
; * 42 = ST_DIG_VID              (0x20 + 0x0A) - 540Mbs
; * 43 = ST_HIGHDEFINITION       (0x20 + 0x0B) - 1.485Gbps smpte292
; * 44 = ST_AUTO_RECLOCK         (0x20 + 0x0C) - reclocks dig vid
; *
; ******note: two other values may be added to the signal type obtained from mcpu
; ******      0x00(0) - ST_SHARED_CONTROL
; ******      0x40(64) - ST_MCPU_EXCLUSIVE_CONTROL
; ****** These values will need to be subtracted off of the signal type and the
; ****** 0x20(32) value added on to get the correct control.
SignalType   = 34
;**********************************
; * Frame Type can be the following:
; * 0  = FT_QUAD_32X32_AUDIO    (0x00)
; * 1  = FT_DUAL_64X64_AUDIO    (0x01)
; * 2  = FT_DUAL_64X32_AUDIO    (0x02)
; * 3  = FT_DUAL_32X64_AUDIO    (0x03)
; * 16 = FT_128X128_VIDEO       (0x10) - used for expanded type
; * 17 = FT_128X96_UPPER_VIDEO  (0x11) - ??
; * 18 = FT_128X96_LOWER_VIDEO  (0x12) - ??
; * 19 = FT_128X64_VIDEO        (0x13)
; * 32 = FT_128X64_AUDIO        (0x20)
; * 34 = FT_AES_NX256           (0x22) - NB256x256, NB512x512
; * 48 = FT_64X64_VIDEO         (0x30) - classic video
; * 64 = FT_32X32_VIDEO         (0x40)
; * 98 = FT_7500WB_256x256      (0x62) - 7500 WB256x256
; * 99 = FT_7500WB_128x128      (0x63) - 7500 WB128x128; *
FrameType = 16
;**********************************
; ConfigOutputSize = the number of outputs per set of Node Controllers
; -Max = 128 (muliples of 32) - Currently must be same for each set!
; (Max for Matrix Controllers is 256)
ConfigOutputSize = 128
;**********************************
; ConfigInputSize = the number of inputs in the router
ConfigInputSize = 256
```

```
;***********************************
; This gives an offset - have not seen it used - maybe with expanded
ConfigOutputOffset = 0
;***********************************
; This gives an offset - have not seen it used - maybe with expanded
ConfigInputOffset = 0
;***********************************
; ExpansionType can be the following:
; * 0 = ET_SINGLE - which means non-expanded
; * 1 = ET_MULTIPLE - which means there are several frames (expanded)
ExpansionType = 1
;***********************************
; Have only seen 0 here so far??
; ONLY FOR EXPANDED SYSTEMS!!
; This appears to be the number of outputs
ExpansionSize = 0
;***********************************
; WhichOutputs will let the NC know which set of up to 128 outputs it is controlling
; Values can be the following:
; * 0   = WO_128X0 - which is the standard for non-expanded. See expanded section below!
WhichOutputs = 0
;***********************************
; NCType lets it know if it is dealing with Standard or Enhanced Node Controllers
; Values can be the following:
; * 0 = Standard node controllers -
; * 1 = Enhanced node controllers
NCType = 1
;***********************************
; This lets the software know which com port to use.
; NCComPortOne is for standard node controllers or the first enhanced
NCComPortOne = 15
;***********************************
; FOR ENHANCED ONLY!
; This lets the software know which com port to use for the second node controller
; If there is only one enhanced node controller - please set to 0.
NCComPortTwo = 16
;***********************************
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
; TRULY EXPANDED _ NEEDS SECOND SET OF NODE CONTROLLERS
; This section is ONLY for TRULY EXPANDED SYSTEMS - OVER 128 OUTPUTS!!
;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
;***********************************
; This value lets the software know if it is expecting a truly expanded system
; >> OVER 128 OUTPUTS!! >>
; UNLESS EXPANDED, PLEASE SET TO ZERO (0)!!!
; * 0 - default and means NO - not expanded
; * 1 - means truly expanded system
WeAreExpanded = 1
;***********************************
; WhichExpandedOutputs lets the NC know which set of up to 128 outputs it is controlling
; Values can be the following:
; UNLESS EXPANDED, PLEASE SET TO ZERO (0)!!!
; * 0   = WO_128X0 - which is the standard for non-expanded.
; * 128 = WO_128X1 - ONLY VALID FOR EXPANDED
; * 256 = WO_128X2 - ONLY VALID FOR EXPANDED - matrix controllers
WhichExpandedOutputs = 128
```

```
;**********************************
; This lets the software know which com port to use.
; NCComPortThree is for standard node controllers or the first enhanced
; UNLESS EXPANDED, PLEASE SET TO ZERO (Ø)!!!
NCComPortThree = 19
;**********************************
; FOR ENHANCED ONLY!
; This lets the software know which com port to use for the second node controller
; If there is only one enhanced node controller - please set to Ø.
NCComPortFour = 17
;///end///
```

Moxa Unit

This drawing shows the pinout of the Moxa converter internal to the NV9500:



In normal circumstances, customers do not need to know anything about this part.

The DIP switch positions shown are required.