Synchronizing a RAVE network to an external clock source
A tutorial

Introduction

The intent of this paper is to provide a guide for configuring a network, via software, when an external clock source is required. The purpose here is to slave the RAVE network to an external system. This may be required when an external system is sensitive to synchronization, has little "pullability" or when the need arises to synch to a secondary audio or video source. Note that this paper discusses synching the network as a *system* to an external source.

Although this tutorial guides the user through the necessary steps to configure a RAVE network implementing an external word clock, it is assumed that the network designer has some experience with the required tools and with the windows OS. This tutorial does not provide details on compiling a MIB or configuring an SNMP MIB browser application. This information is located on the QSC Audio website at the RAVE technology papers location.

Required tools and products

To perform the procedures outlined in this document a number of tools, software and hardware products are required.

First, it is assumed that RAVE products are used within the network configuration. Strictly speaking, the majority of information in this tutorial *may* be applied to other CobraNet hardware products. However, the methods described here have only been tested at QSC Audio with RAVE products.

A utility application is needed for assigning IP addresses to the RAVEs. A BOOTP server or the CobraNet Discovery application may be used. This paper uses the CobraNet Discovery application in its examples.

The configuration of RAVEs is achieved by altering MI (management interface) variables from their default state. Most configuration requirements can be met by front panel hexadecimal switch selection and through proper connections on the rear panel of the RAVE products. However, this paper discusses altering MI variables via software using the SNMP (simple network management protocol) method. This requires some supporting items.

1) An SNMP agent must be installed within the RAVE. For the RAVE products, the SNMP agent is shipped with the CobraNet code in versions 2.6.9 and beyond. CobraNet version 2.8.5 or beyond is required for step 6 in this tutorial, which configures the persistence of management variables.

2) At the computer console, there must be an application that supports writing to SNMP object locations. The QSC Audio website has a link to MG-Soft, which provides a trial version of an SNMP browser application. This application is used for all SNMP related screen shots in this paper. Other SNMP applications may be used, although the methods may differ from the procedures outlined here.

3) The SNMP application requires something called a MIB (management information base) that references the locations specified in the agent and describes the variables. Peak Audio has a MIB of CobraNet variables available for download from their website. The most recent version of the "peakAudio" MIB should be used. As of this printing, the most recent revision of the MIB is dated 03/26/01. The MIB can usually be viewed with a text editor or word processor such as "Notepad". The date can be seen in the revision history at the top of the MIB file. The version number should be 27 but has not been updated recently.

Links to all the necessary tools are located on the QSC Audio website from the RAVE page. Peak Audio has many of these tools available for free download directly from their own website. QSC has also made other reference papers available that describe compiling a MIB and getting started with RAVE software configuration via SNMP.

Note: CN Discovery requires the Windows 98, Windows NT or Windows 2000 operating systems. Discovery is not supported for use with Windows 95 or 3.1.

The widget network

For simplicity this network will consist only of four RAVE devices, a PC and a "simple switcher" or unmanaged network switch. The RAVEs consist of a 188s, 88s, 160s and 161s. The network switch is a Hewlett Packard 408, 8-port, 10/100 auto-sensing switch. All connections use CAT-5 cable. The management console, or PC, is a basic Win 98 machine with an Intel Pro 100+ network adapter, which is assigned an IP address of 191.34.150.37. The SNMP configuration procedure is the same for all RAVE units, although we will setup the 160s for external synch and the conductor role.

Clock issues

The RAVE family of products provide the ability to drive the 48 kHz wordclock input, labeled as slave input on the rear panel of the RAVE device, from an external clock source. Additionally, the RAVEs provide the ability to multiply a low frequency external clock source to the required rate.

The external clock requirements are such that the clock source must be a multiple of the CobraNet™ cycle rate and that the accuracy of the clock be equal or better than +/- 50 ppm. Since the current cycle rate is fixed at 64 sample periods, the clock source must be a multiple of 750 Hz.

1/48000 Hz • 64 samples ≅ 1.333 ms   This is the cycle period (T)
1/T = *f*   1/1.333ms = 750 Hz   This is the base unit that the RAVE's internal clock can lock to.

An external clock rate of 33750 Hz is valid (45 • 750 Hz = 33750 Hz). Note that some external clock rates of common frequencies, such as 44.1 kHz and 32 kHz, are not supported.

Accuracy of +/- 50 ppm (parts per million) requires that the external clock source have a precision of at least +/- 2.4 Hz at a 48 kHz rate (48000 ÷ 1/1000000 • 50 = 2.4). Note that it becomes increasingly difficult to supply an accurate clock at low rates (ex. at 750 Hz the clock must be accurate to within +/- .0375 Hz).
In addition, the clock must not drift more than +/- ¼ sample period or +/- 5.2 µs.

Clock specifications for all CobraNet hardware platforms are detailed in the CobraNet datasheet.

Procedure outline

In order to provide synchronization to an external clock source with a RAVE network, we must assure that the device distributing the network clock is predictable. The clock source on a CobraNet network is the conductor unit. RAVE products indicate which unit is acting as the network conductor by illuminating a yellow LED on the front panel of the incumbent unit. CobraNet assigns the conductor role based primarily on a priority system and secondarily on hardware or MAC address. The RAVE priority rating is such that the 160s/80s device models have the highest conductor priority rating due to their low transmission requirements. The RAVE 188s/88s models hold the middle priority rating and the 161s/81s models are the least likely to assume the conductor role. It is recommended that the RAVE 160s or 80s devices be used when a permanent conductor or external clock source is needed.

The procedure to setup a RAVE network system implementing an external clock source will include the following steps:

1) Assign an IP address to the intended RAVE conductor.

2) Open the "peakAudio" MIB in an SNMP MIB browser, address and establish communication with the intended RAVE device.

3) Set the transmit and/or receive bundle assignments, depending on the model's audio I/O.

4) Set the necessary MI variable for implementing an external clock source.

5) Set the conductor priority.

6) Enable the persistence of these variables so that the RAVE will retain all configuration settings upon power cycle. ** This step is optional.

After completing software configuration of the network "conductor", the remaining RAVE "performers" will require setup of the appropriate bundle assignments via the hexadecimal switches (or via SNMP if desired). The clock source can then be connected to the rear panel "slave input" BNC connector on the RAVE conductor.

Step 1

Open the CobraNet Discovery (Disco) utility or your BOOTP server application. The CobraNet Discovery application should be setup so that the RAVE network and the PC running the application are on the same subnet.

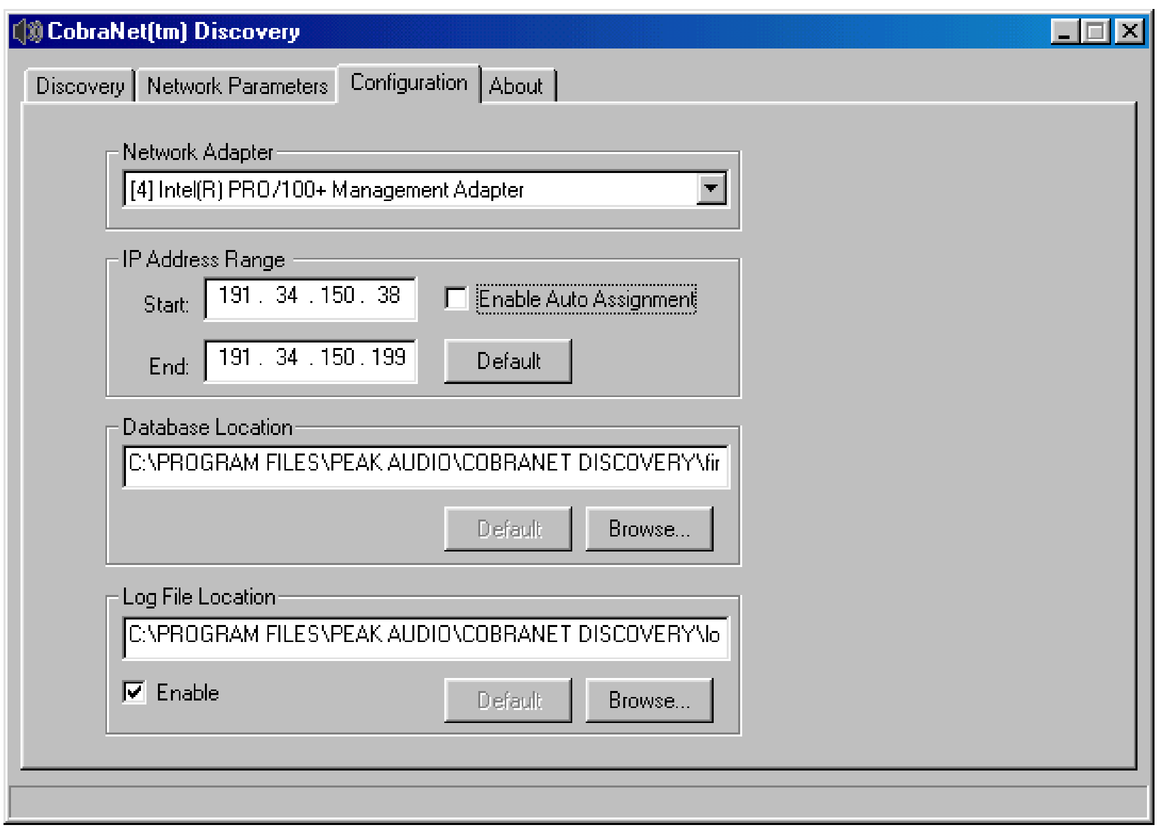

Figure 1.

Typically, the system is setup for class "C" networks. In this paper we deviate slightly from the normal IP address numbering scheme. The first octet (191) is used to distinguish the RAVE network from our in-house data communications network. Note that in the configuration pane of figure 1, we have selected the appropriate network adapter for the network (it is important to select the correct NIC connected to the RAVE network, as there may be multiple adapters installed). Also, we have selected the range of IP addresses to be assigned from 191.34.150.38 to 191.34.150.199.

"Enable Auto Assignment" is unchecked to show the IP address range. We will use the auto assignment feature for the actual delegation of IP addresses. Therefore, this box should be checked after the address range is entered. Note that the range of IP addresses may be automatically filled in if Disco detects your network adapter's IP address. In either case, the adapter and RAVE's must be on the same subnet.

If you're using a BOOTP server the configuration of your application will obviously be quite different. Typically, BOOTP servers also allow you to enter the IP address manually or by allocating a range of addresses. However, the BOOTP server will likely not be able to distinguish between CobraNet devices and other network hardware. You may also be required to specifically state that there is no "boot" file to be uploaded to the RAVEs. We're simply using the server to prepare the device for remote "boot" by assignment of an IP address. The IP address is all we need from the server. Refer to RFCs 951 and 1542 or your BOOTP server application documentation for more information on setup and BOOTP procedures. An example of a BOOTP server application is shown in figure 2.

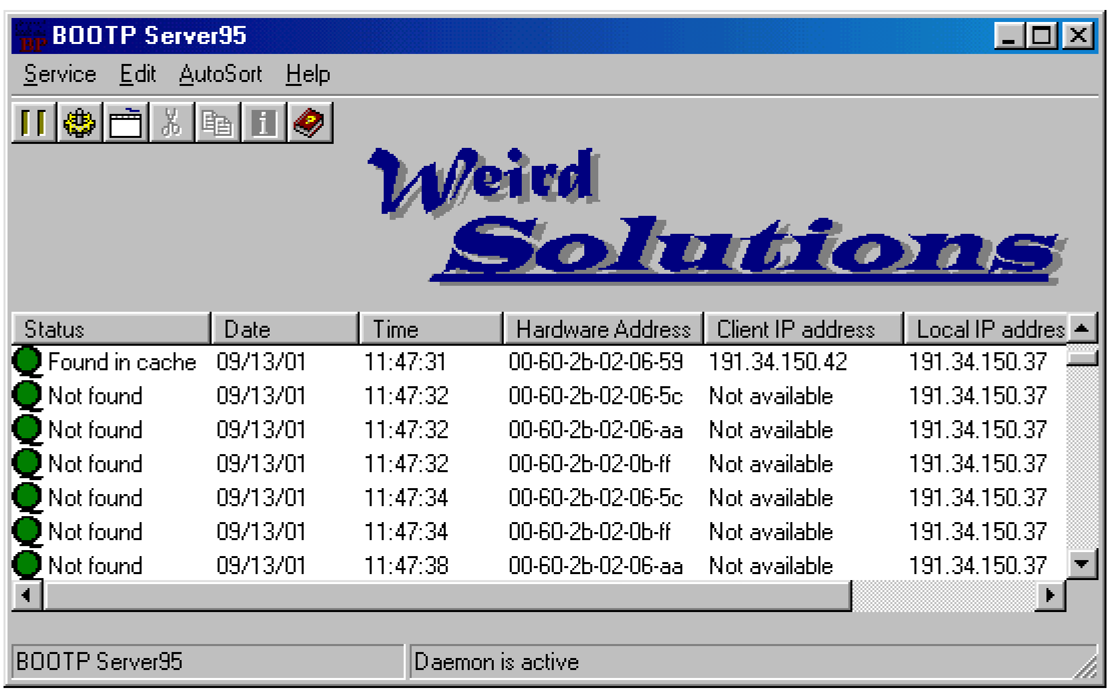

Figure 2.

In the example shown in figure 2, only the RAVE 160s is connected to the network. This makes it a little easier to see what is taking place. The BOOTP server is polling for BOOTP requests from known clients, such as the RAVE 160s and 161s devices. When the client's request is received, the server assigns the appropriate IP address as specified in the application's configuration parameters. In this case, IP address 191.34.150.42 is assigned to the RAVE 160s. This can be seen on the first status line. For reference, we can compare this line to Disco's MAC to IP association for the 160s later in the tutorial.

The remainder of this tutorial will assume that we are using the CobraNet Discovery application and the tools as outlined in the "Required tools and products" section above.

Refer to the CobraNet Discovery manual for complete details on installing and configuring the Disco application for use with your network adapter(s). The Discovery manual is available for free download from the Peak Audio website.

Note: CobraNet Discovery requires the Windows 98, Windows NT or Windows 2000 operating systems. At the time of this printing, CobraNet Discovery is not supported for use with Windows 95 or 3.1.

Figure 3 shows the Discovery pane and the physical, or MAC, address of four RAVE devices.

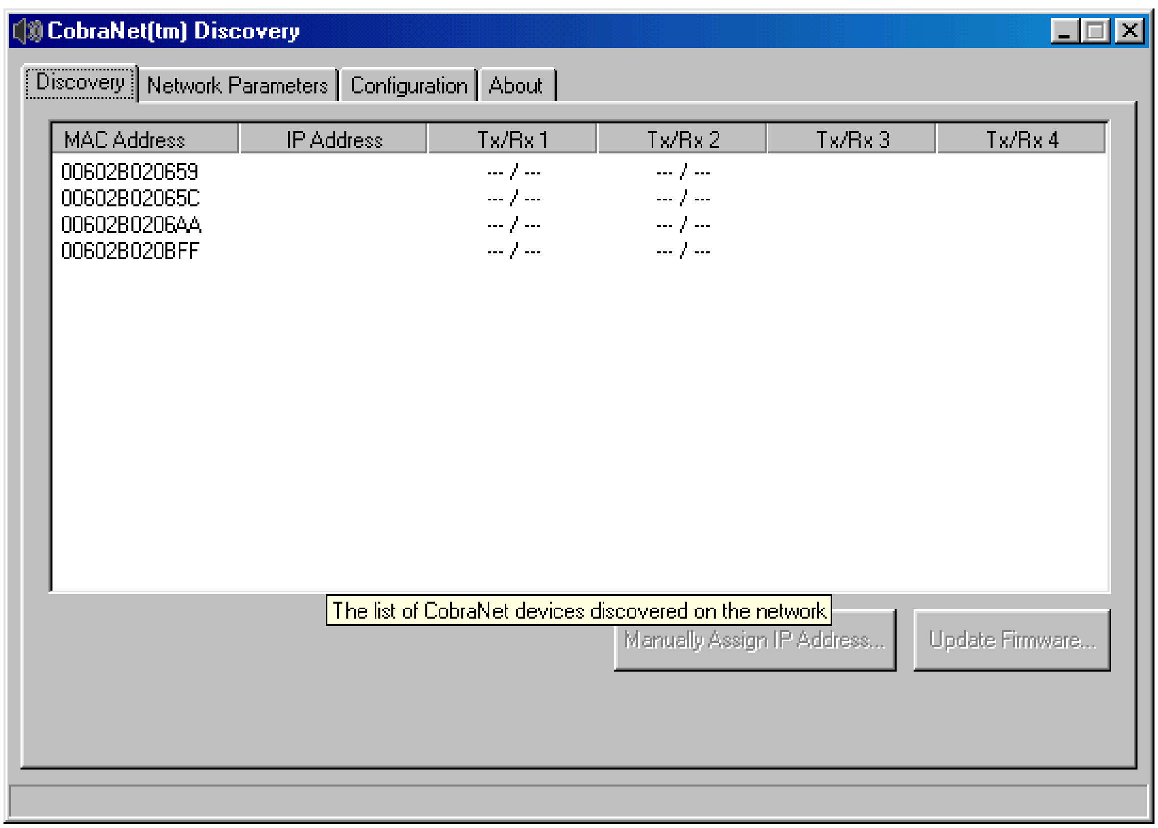

Figure 3.

Prior to the network configuration we have set all RAVE hexadecimal switches to zero. Therefore, the Tx/Rx bundle assignments indicate a "defeat" state (···).

Note that each RAVE has two bundles for transmit and two bundles for receive, which are shown on the Discovery window. This is the true configuration of the RAVE core regardless of whether there is audio I/O available to support all bundles.

Remember that we unchecked the "Enable Auto Assignment" box on the configuration pane. This was done so that we could view and assign the range of IP addresses. If we check this box we should see that Disco has assigned an IP address to each of the RAVE devices. This is shown on the Discovery pane as indicated in figure 4.

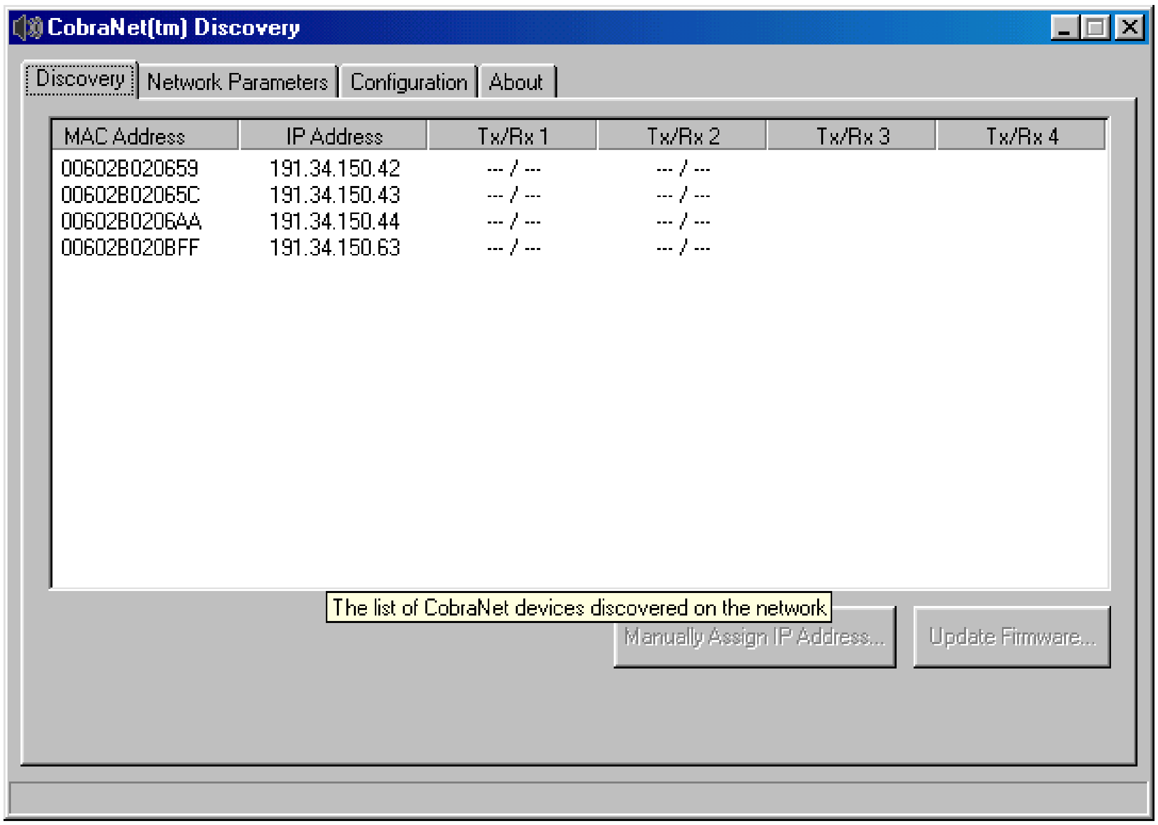

Figure 4.

Step 2

To begin configuring the RAVE devices we need to open the SNMP browser application. The browser will allow us to "get" variables to view the current status or value and to "set" variables to the desired value or enable features in the MI.

The SNMP browser's primary pane is the query window. This window shows the tree structure of the loaded MIB modules. This is indicated in figure 5. In this example we have not loaded any MIBs other than the "peakAudio" MIB (Refer to the MIB compiler application or the QSC website for compiling a MIB module and getting started with SNMP). We can see that the standard SNMP hierarchy is present along with CobraNet's "coreManager" folders, which group each family of associated MI variables.

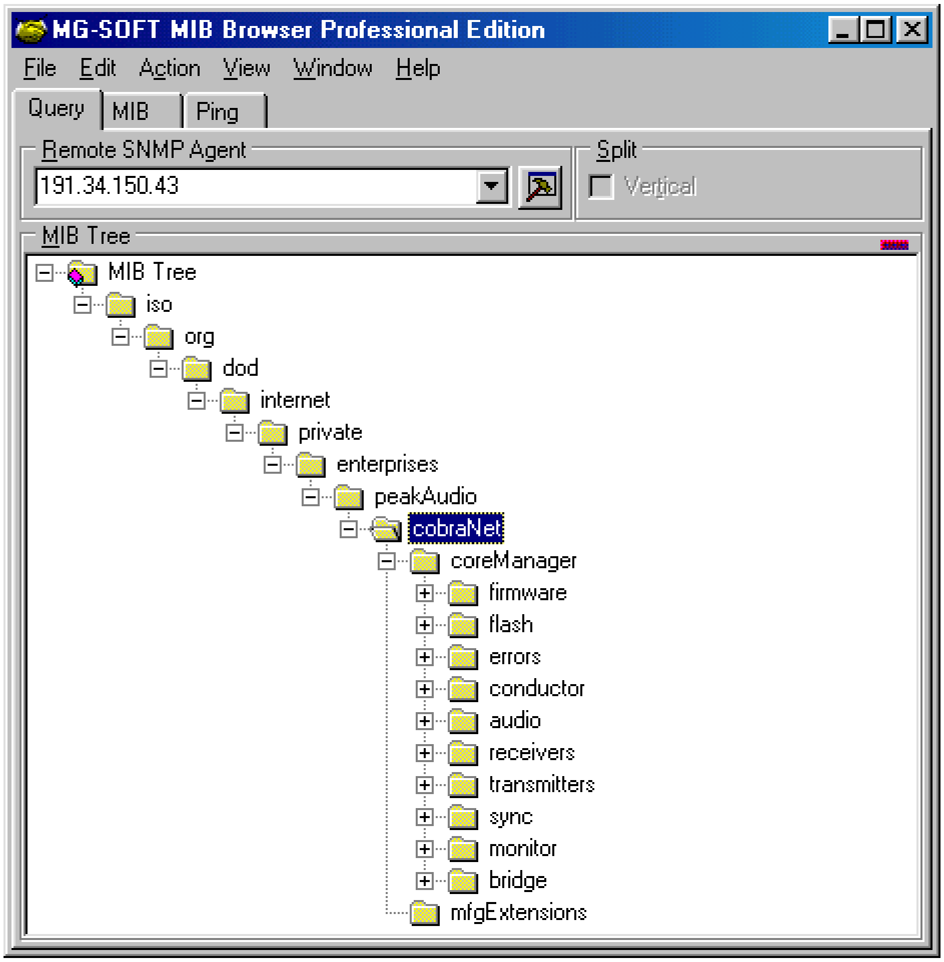

Figure 5.

The IP address for each RAVE is entered in the "Remote SNMP Agent" dialog box. This particular application allows you to access all IP addresses from each subfolder as well. In figure 5, the IP address for the RAVE 161s is shown. How do we know? We can obtain the identity of the device in a number of ways. We can determine the device by front panel clues (we normally label the units with their last IP octet), we can use Disco and read the boot bank of the device or we can use SNMP and "get" the "firmwareMfgProductId" variable or "get" the MIB-II "sysDescr" variable. The MIB-II "sysDescr" variable is the easiest and most complete description. This variable indicates the manufacturer, product and CobraNet firmware for the device. It's also possible to assign a system name via SNMP text entry in the “sysName” variable or through the RAVE’s front panel hexadecimal switches when “flashPersistEnable” is set to a non-zero value. Configuring persistence will be decribed later in this tutorial.

If we click on the hammer icon next to the “Remote SNMP Agent” dialog box, the configuration parameters for the MG-Soft application appears. This is shown in figure 6. Different releases of the MG-Soft trial software appear slightly different. With this release we need to set the SNMP protocol version, community “set” string, timeout, number of times to retransmit a lost or unverified message, and the SNMP port number.

The SNMP protocol version radio button should always be set to version 1 (SNMPv1).

The community string should be set to private. This means that we cannot alter variables outside of the private community (refer to related SNMP RFCs or application help files for additional info). Note that some MG-Soft releases and other SNMP applications allow the user to set both the read and write community. If the read community must be configured, you will likely want to set this parameter to public. This allows access to MIB-II variable “get” requests and RMON information when the appropriate MIBs are loaded.

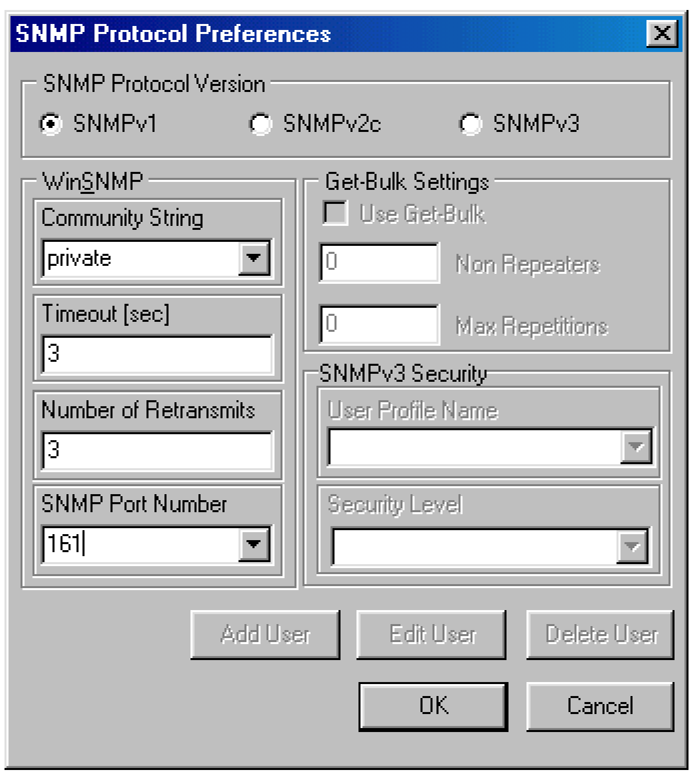

Figure 6.

The timeout and retransmit parameters can be set to any practical value. Setting the timeout equal to 3 seconds here is just for demonstration, as this is a small network.

Usually the response should be fairly quick. A value of 1 second is more realistic for a small network. The number of retransmits should be limited. More retransmissions causes more network traffic and can eat bandwidth at the interface. Normally you'll want to achieve a balance of retransmits to the timeout period. This is especially true if the network is large. Allowing a longer response time with fewer retransmits is preferable. Reducing the timeout to a short period with a large retransmit number can halt a network.

Since CobraNet does not support traps, the SNMP port number should always be set to 161.

Step 3

We now want to set the receive bundles of the RAVE 160s. The RAVE 160s is assigned the IP address 191.34.150.42. We need to enter this address in the "Remote SNMP Agent" dialog box.

We can then open the "receivers" folder beneath coreManager within the query window. The subfolders "rxTable" and "rxEntry" should also be opened. Right mouse click on the "rxChannel" leaf. This is shown in figure 7.

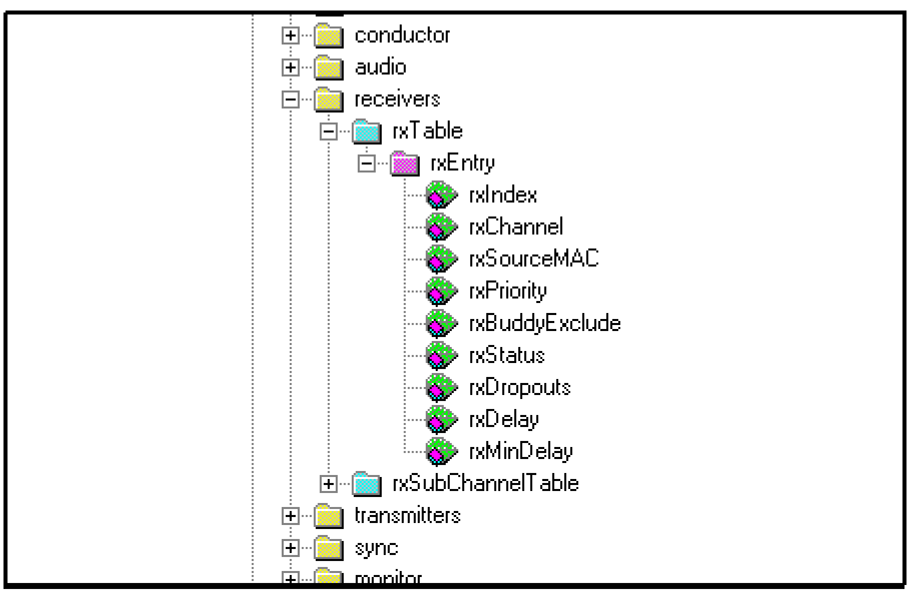

Figure 7.

Note that CobraNet terminology for network "channels" has been changed to "bundles". Depending on the MIB version in use, "rxChannel" or "rxBundle" may appear as the leaf name. The CobraNet datasheet has been updated and replaces references to network channels with the term bundles.

In the drop down selection choose “info”. If the SNMP MIB browser or related application version supports instanciation, we can select “no instance” for the purposes of this configuration setup. The bundle assignments for the unit should now be visible. This is the background window shown in figure 8. Note that the selected device’s IP address is shown in the upper dialog box. The current “value” or assignment of bundles will be shown in the main window. If we have previously set the RAVE hexadecimal switches to zero, then we should see “0” as the initial value for both “rxChannel.1” and “rxChannel.2” when the window first appears.

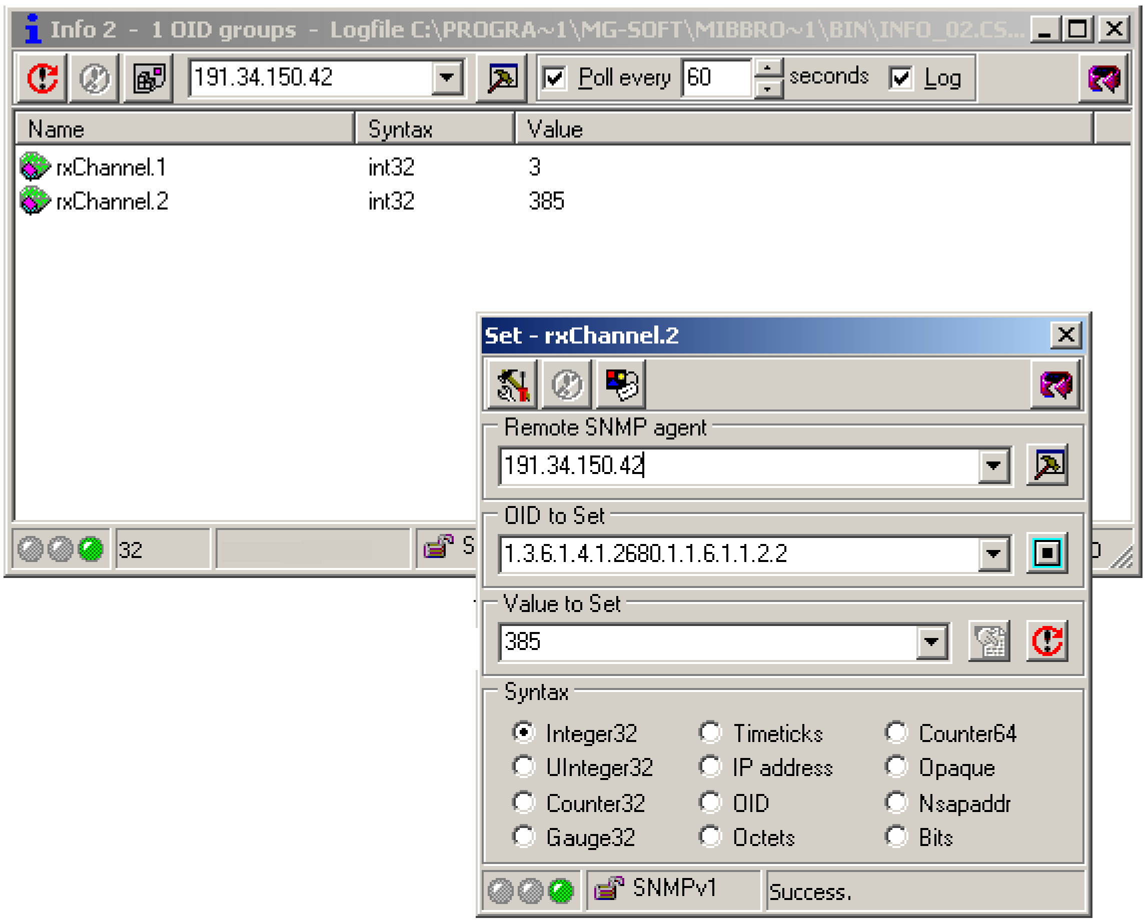

Figure 8.

You can verify the SNMP setup parameters by again selecting the hammer icon.

We now want to right click on “rxChannel.1” and select “set” from the drop down menu. The “set” window should now be visible, as shown in the foreground of figure 8. The variable reference or object identifier (OID) is filled in automatically for us. Note that with older versions of this SNMP browser you may have to fill in the SNMP set parameters the first time the application is opened. This is done by selecting the hammer icon on the "set" window.

We can choose a multicast address for the first bundle by entering the value 3 in the "Value to Set" dialog box. The syntax is filled in automatically, but should be set to integer32. Once the multicast assignment is entered, we can select the tools icon (hammer, screwdriver and wrench). The window should then show the green indicator at the bottom of the screen and indicate success at the bottom right. Depending on the polling rate, the "info" window may not show the update immediately. However, you can verify the variable status by selecting the refresh button in the info window. This is the top left button with the pointed red semi-circle and centered exclamation mark. It's a good idea to keep the polling rate fairly low, especially if you're running additional SNMP based applications.

We can alter the bundle assignment for "rxChannel.2" in the same way. In this example, we have selected unicast bundle 385 for the second entry. CobraNet unicast assignment 385 is the equivalent of setting the RAVE's front panel hexadecimal switches to "81" (refer to RAVE bundle assignments on the QSC or Peak Audio websites). The SNMP configuration procedure is the same as for "rxChannel.1". Figure 8 shows the resulting values and the success indicator for the unicast bundle.

Step 4

Now we'll set the necessary variables for using an external clock source with the RAVE 160s. To do this we can close the "receivers" folders and open the "sync" folder beneath coreManager. Right click the "syncConductorClock" leaf. The "syncConductorClock" leaf and all leafs beneath the "sync" folder are shown in figure 9.

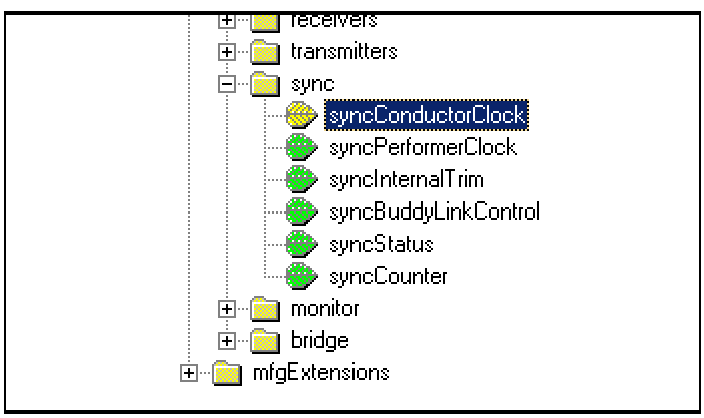

Figure 9.

Select "properties" from the drop down menu. The properties window shows the valid values for this variable and a description of the clock modes that they enable. In most cases, the VCXO is parked when the RAVE is acting as the system conductor. This indicates that the unit is using its own internal VCXO to source the clock. Instead, we want to enable “EXTWRDCLK” mode so the RAVE 160s unit will synchronize to and distribute an external clock. This mode requires that we alter the “syncConductorClock” variable to the value 1.

To enable “EXTWRDCLK” we need to exit the properties window, again right click on the “syncConductorClock” leaf and select “info” from the drop down menu. The resulting info window should indicate the current value of the variable. This value should initially be set to RAVE’s default state of 0.

If we right click on the variable and select “set” from the drop down menu, we should see a “set” window with properties similar to the one shown in figure 8. Only the OID should be different. Enter the value 1 in the “Value to Set” dialog box and initiate the change by selecting the tools icon at the top of the set window. Upon refresh, the result in the info window should be similar to that shown in figure 10.

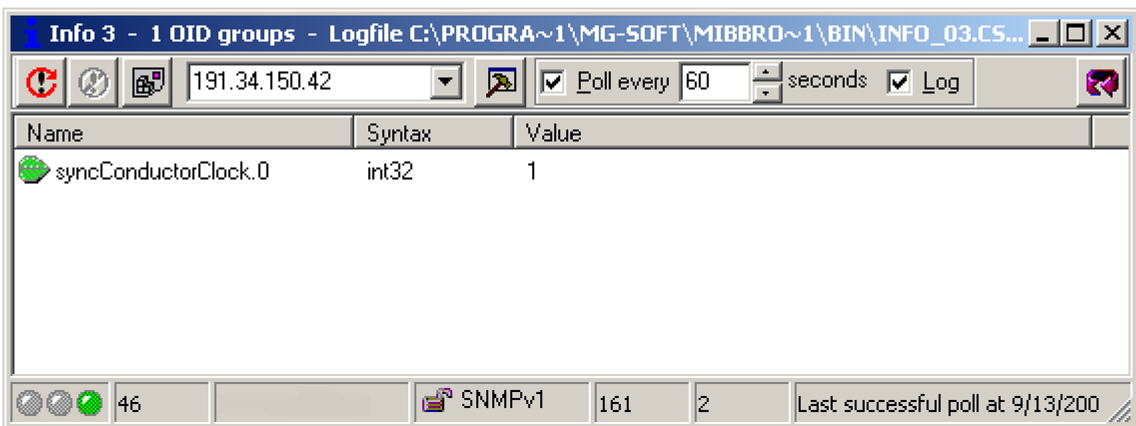

Figure 10.

Step 5

In order to assure that the RAVE 160s will retain the conductor role and that the network will slave to an external clock, we need to elevate the unit’s conductor priority level. We may also choose to lower or null the priority of all RAVE performers, although this requires configuration of all units via SNMP.

We can be reasonably assured that the RAVE 160s will always assume the conductor role if we assign it the value 30. The 160s already holds the highest conductor priority rating for the RAVE family of products. However, the default rating is assigned the value 18. Elevating this priority rating to 30 assures that it will have hierarchy over all other

RAVEs in their default configurations. Note that other CobraNet products may have conductor priority ratings higher than 30.

To set the conductor priority rating we need to close the "sync" folder and open the "conductor" folder beneath the coreManager.

Right click on the "condPriority" leaf. Select "info" from the drop down menu. We should see that the default value for this variable is 18. Right click on the "condPriority.0" OID and select "set" from the drop down menu in order to invoke the "set" window. Enter the value 30 in the "Value to Set" dialog box and select the tools icon at the top of the window. Upon refresh, the result should be shown in the info window as presented in figure 11.

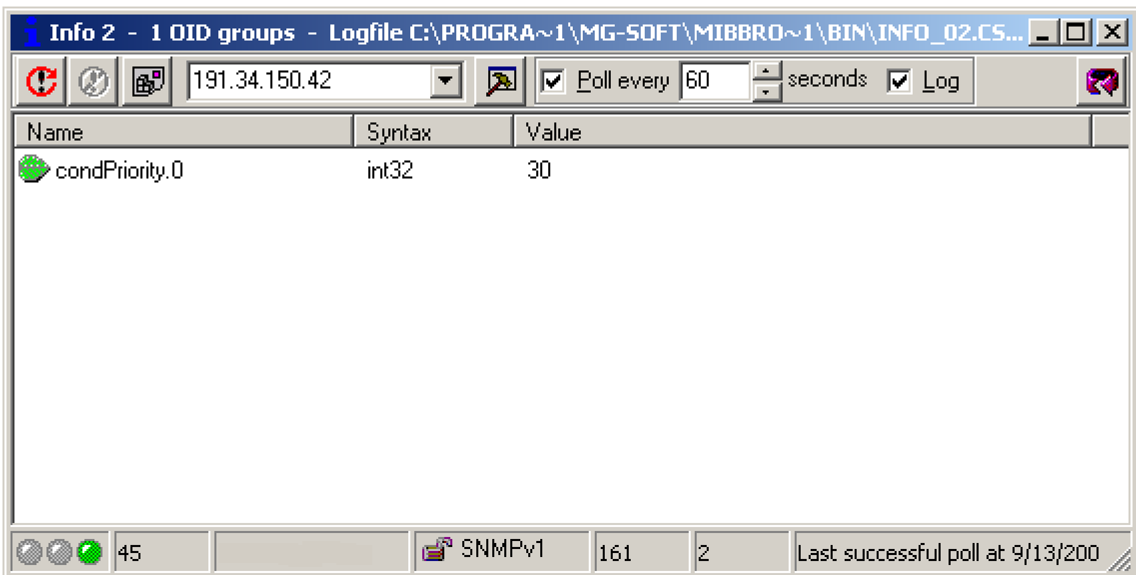

Figure 11.

Step 6

As we progress through the tutorial, the repetitive information is reduced for each step. However, the method for accessing variables and menu options is the same. Note also that we do not need to go to the info window for each transaction. We can "set" variables from the leaf location. However, when working with the SNMP interface for the first few times it may be beneficial to see what the status of the variables are prior to altering them.

In step 6 we want to enable the persistence of our variables so that we do not have to repeat the entire RAVE 160s device configuration the first time a power cycle or reset event is encountered. Persistence provides permanent memory storage of some variables. Refer to the CobraNet datasheet for a list of persistence capable variables.

*** Note that step 6 requires CobraNet version 2.8.5 or above installed in the RAVEs.*

Close the "conductor" folder and open the "flash" folder beneath coreManager.

Right click on the "flashPersistEnable" leaf and select "info" from the drop down menu. We should see the info window with RAVE's default state of 0 for the "flashPeristEnable.0" OID.

In order to invoke persistence of the MI configuration we need to alter this variable's value to 1. Note that when persistence is enabled software control is assumed. Because of this, RAVE's front panel hexadecimal switches are no longer available for bundle assignments or MI variable selection. Instead, the front panel switches act as the "sysName" selection for the device. This was referenced earlier in the document in regards to MIB-II variables. Note that software control can be aborted by setting all hexadecimal switches to "F" and power cycling the RAVE.

Continuing, right click on the "flashPersistEnable.0" variable name and select "set" from the drop down menu. In the set window's "Value to Set" dialog box enter the value 1 and select the tools icon at the top of the window to invoke the change. Upon refresh, the info window should look like the image in figure 12.

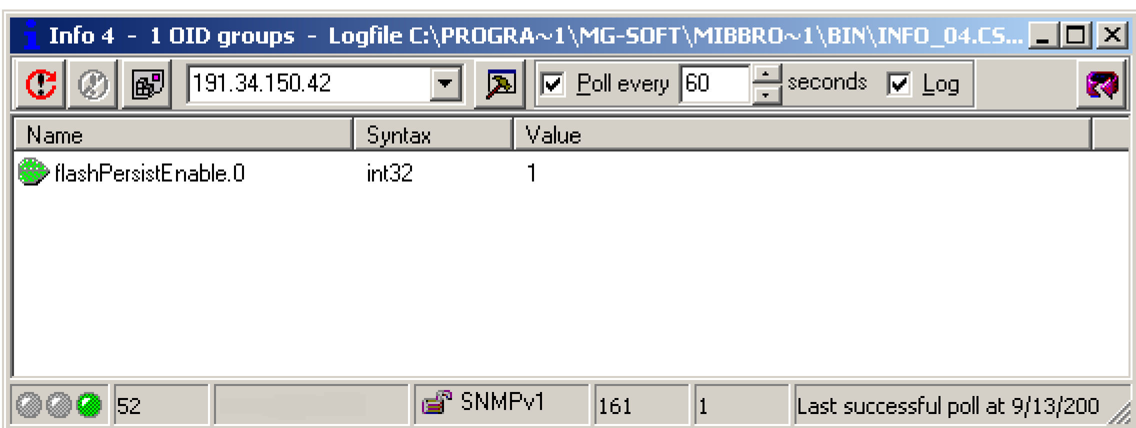

Figure 12.

It's recommended that the "flashPersistAck" variable be set to a non-zero value in order to force a write of all persistent variables to internal memory. Otherwise, the variable set will be written during the next scheduled "set" event. This is just a precautionary step. This can be performed by selecting the "flashPersistAck" variable and performing the steps as described in "Step 6".

Verification

Configure the RAVE performers through the front panel hexadecimal switches or SNMP to communicate with one another as desired. Those performers communicating with the RAVE 160s unit must have bundle assignment selections that mate with the assignments configured in this tutorial.

After connecting the external clock source to the RAVE 160s, verify that there are no communications errors or audio dropouts. Synchronization errors are indicated on the front panel of the RAVEs when both the “txError” and “rxError” LEDs are illuminated together, either flashing or steady. If error displays are seen, verify the following conditions:

- Make sure that the external clock source is valid and meets the specifications as outlined in the CobraNet datasheet. This is referenced at the beginning of this document.

- Make sure that the amplitude of the clock meets standard HCMOS levels for +5 volt logic. The word clock input on the RAVEs require an amplitude level of +3.5 volts on the positive transistion to assure meeting component logic requirements.

- Make sure the cables are high quality and properly terminated. Refer to the Belden specs for coax cable terminations and length for the intended clock rate. RAVE products use 50 ohm BNC connectors.

- *Verify the configuration of variables as outlined in this document*.