MMB Networks™

RapidHA Desktop User Guide v1.2.0

MMB Networks
25 Prince Arthur Avenue
Toronto, Ontario, Canada
M5R 1B2
(416) 636-3145

Legal Notice

THIS DOCUMENT (“DOCUMENTATION”) CONTAINS HIGHLY SENSITIVE, CONFIDENTIAL, AND PROPRIETARY INFORMATION OF MMB RESEARCH INC. (“MMB”) AND MAY BE THE SUBJECT OF PATENT APPLICATIONS.

THE DOCUMENTATION IS PROVIDED STRICTLY SUBJECT TO THE TERMS AND CONDITIONS OF THE MMB SERIAL PROTOCOL DOCUMENTATION LICENSE AGREEMENT ("AGREEMENT") INCLUDED IN THIS PACKAGE. THE AGREEMENT IS A LEGALLY BINDING AGREEMENT BETWEEN YOU (EITHER AN INDIVIDUAL OR ENTITY, AS APPLICABLE) ("YOU" OR "YOUR") AND MMB AND IS ENTERED INTO FOR GOOD AND VALUABLE CONSIDERATION.

PLEASE READ THE TERMS AND CONDITIONS OF THE AGREEMENT CAREFULLY BEFORE DOWNLOADING, READING OR USING THE DOCUMENTATION. THEY DEFINE YOUR RIGHTS AND OBLIGATIONS WITH RESPECT TO THE DOCUMENTATION. ANY USE OF THE DOCUMENTATION WILL BE DEEEMED AS YOUR COMPLETE AND IRREVOCABLE ACCEPTANCE OF ALL THE TERMS AND CONDITIONS OF THE AGREEMENT.

IF YOU DO NOT AGREE WITH ANY OF THE PROVISIONS OF THIS AGREEMENT, DO NOT DOWNLOAD, READ, COPY OR USE THE DOCUMENTATION, AND IMMEDIATELY RETURN ALL HARD COPIES OF THE DOCUMENTATION AND DELETE ALL ELECTRONIC COPIES, WHETHER UNDER YOUR CONTROL OR POSSESSION, AND NOTIFY MMB IN WRITING OF YOUR COMPLIANCE THEREWITH.

Table of Contents

1.0 Introduction

RapidHA Desktop is an easy-to-use software tool for anyone who wants to quickly add ZigBee Home Automation (HA) control to a new or existing product. With RapidHA Desktop you can exercise the ZigBee functionality of RapidConnect: the embedded firmware operating on the radio module contained in the RapidHA USB stick.

1.1 New Features

RapidHA Desktop v1.1.0 adds support for the following:

- Serial Firmware Upgrade, see section 4.3
- OTA (Over The Air) Firmware Upgrade, see section 6.5
- Real-Time Clock Interface, see section 6.4

2.0 System Requirements

For successful operation of the software, the following is required:

- PC running Windows 7 or later
- Java Version 7 or later – a PC running a 64-bit operating system requires 64-bit Java
- Com port for each RapidConnect USB stick in use

3.0 Software Installation

The RapidHA Desktop application requires two separate software installations:

1. USB Stick Device Driver: download the device driver zip file for the RapidConnect USB stick at this location: http://mmbnetworks.com/downloads/ Double click on "setup.exe", located in the downloaded zip file.
2. RapidHA Desktop Installer: double click on the installer file sent to you and follow the wizard. If you do not have the RapidHA Desktop Installer, please email support@mmbnetworks.com

4.0 Run the RapidHA Desktop Application

This section documents the steps needed to start the desktop application and communicate with the USB stick.

Connect Hardware: Plug the USB sticks into the computer(s) on which the RapidHA Desktop was installed.

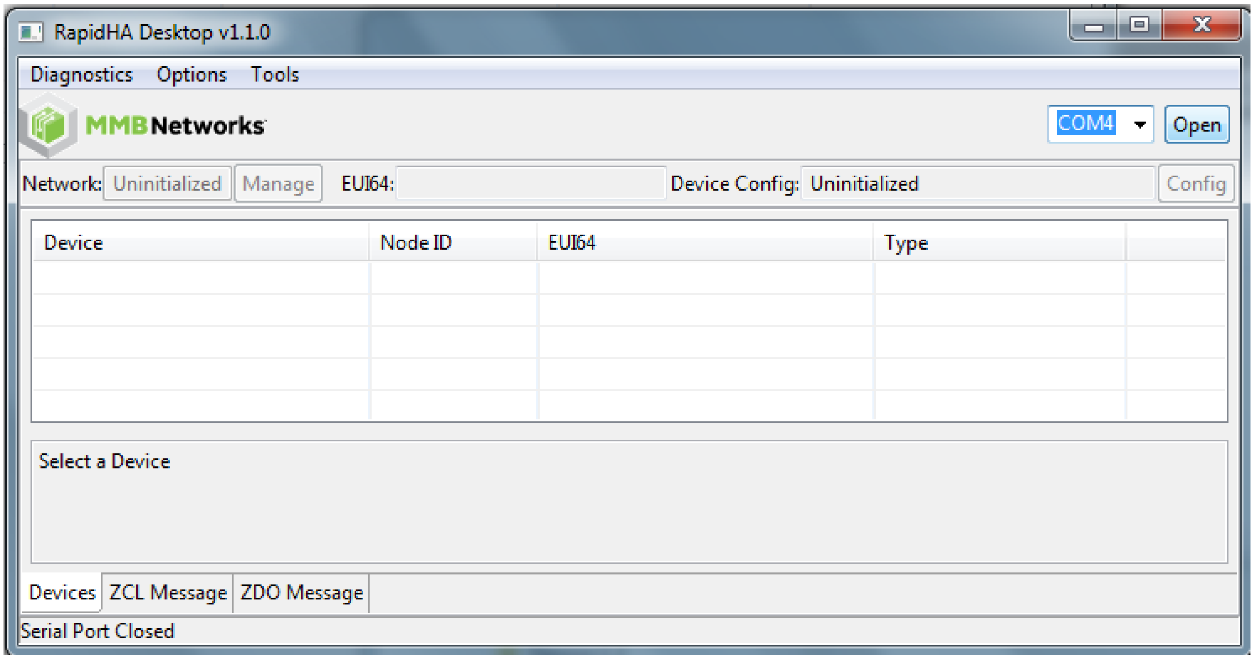
Launch Software: Open and run two instances of RapidHA Desktop by double-clicking on the desktop icon created by the installation process.

Open Serial Ports: Within each program instance, click on the button labeled "Open" which is located in the upper right hand corner of the program window. See Figure 1 to view a screenshot of the initial program window.

Configure ZigBee Devices: Within each program instance, click the button labeled "Config" that is located in the upper right hand corner of the program window. The Device Configuration popup window will appear (see Figure 2 to view a screenshot). From this popup window, click on “Select” to open a list of configuration files.

One USB stick must be configured as the ZigBee coordinator. This is done by selecting and opening the file “coordinator.xml”. The second RapidConnect USB stick must be configured using one of the other XML files.

Figure 1: Opening Screen of RapidHA Desktop



COM4

Network: Uninitialized

EUI64:

Device Config: Uninitialized

Device	Node ID	EUI64	Type

Select a Device

Serial Port Closed

The RapidHA Desktop program always begins execution with the serial port closed and with the “Devices” tab selected. This information is located along the bottom of the program window as shown in Figure 1.

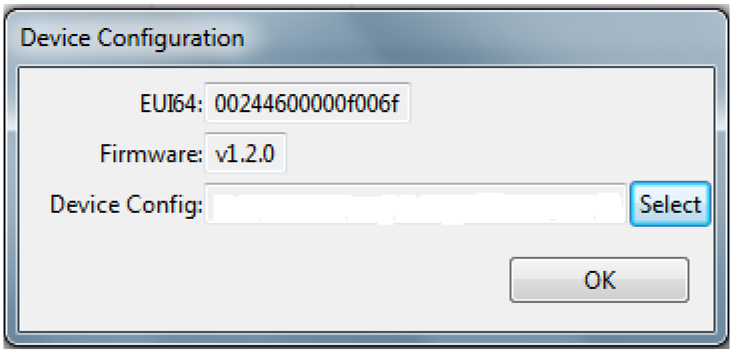
The serial port must be open for the RapidHA Desktop to communicate with the USB stick. The program remembers its last network state and returns to it when the serial port is opened by clicking the button in the upper right corner labeled “Open”.

4.1 Initial Program Execution

The first time the program executes, the device configuration will be “Uninitialized” This information is located near the top of the program window on the right side. As soon as the serial port is opened, the button labeled “Config” will become active to allow device configuration. Click on this button to bring up the Device Configuration popup window shown in Figure 2.

Click the button labeled “Select” to browse for one of the configuration files described in Section 4.2. Use the most up-to-date version of these files in order to access the full cluster support available.

Figure 2: Device Configuration Popup Window



Device Configuration

EUI64: 00244600000f006f

Firmware: v1.2.0

Device Config:

The Device Configuration popup window is only available when the corresponding USB stick is not part of a network.

4.2 Configuration Files

A configuration file contains the following:

- Physical Device Properties – this defines whether the device is a Reduced Function Device (RFD) or a Full Function Device (FFD) and if said device has a sleepy mode or not.
- Application Profile (profileId) – this is 0x0104 for home automation.
- Device Identification (deviceId) – this number is found in the ZigBee Home Automation Public Application Profile.
- Endpoint, Cluster and Attribute Information –a minimum of one endpoint must be defined. The cluster IDs are enumerated in the ZigBee Cluster Library (ZCL) specification.

The RapidHA Desktop installation includes the download of several different configuration files. Only the one named coordinator.xml should be used to form a network.

At the time of this writing, the RapidHA Desktop installation includes the following configuration files:

- coordinator.xml – configures the device to be a ZigBee coordinator, with client clusters defined for all the server clusters defined in the other configuration files.
- doorlock.xml – example of the Door Lock cluster.
- level_controllable_output.xml – example of On/Off cluster and Level Control cluster.
- power_outlets.xml – example of multiple endpoints on one node
- thermostat.xml – example of the Thermostat cluster.

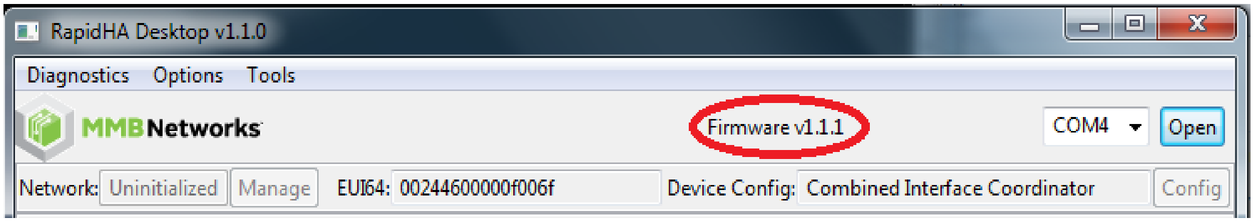
4.3 Firmware Upgrade

There are two methods available for upgrading the RapidHA firmware via the RapidHA Desktop program: serial bootload and OTA bootload. The serial method works only when the network is down. Conversely, the OTA bootload works only when the network is up.

The serial bootload is described here and the OTA bootload is described in Section 6.5.

To check the current firmware version running on a device, open its serial port and the information will be displayed at the top of the program window as shown in Figure 3.

Figure 3: Firmware Version



Firmware v1.1.1

COM4

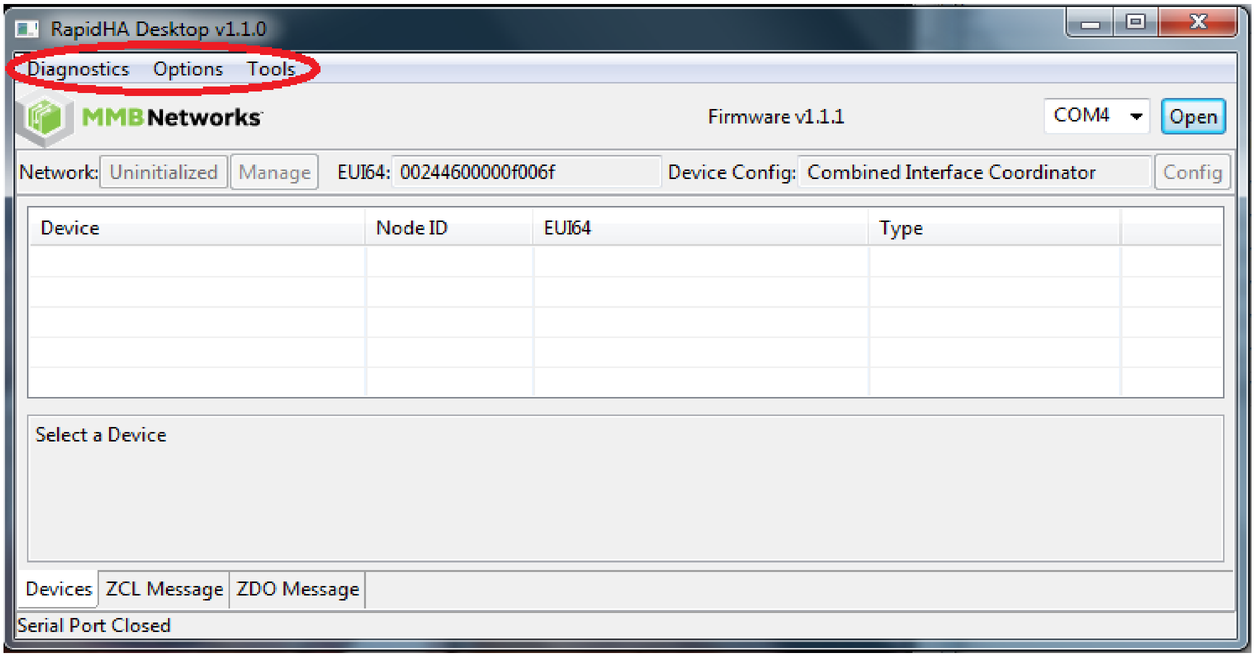
Network: Uninitialized

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

The location of the command to serially upgrade the firmware is located in the upper left area of the program window, as shown in Figure 4. Click on the button labeled “Tools” to reveal a dropdown menu. Select the option “Upload Application” to see the popup window pictured in Figure 5.

Figure 4: Location of Serial Upgrade Command and Other Diagnostic Operations



Firmware v1.1.1

COM4

Network: Uninitialized

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type

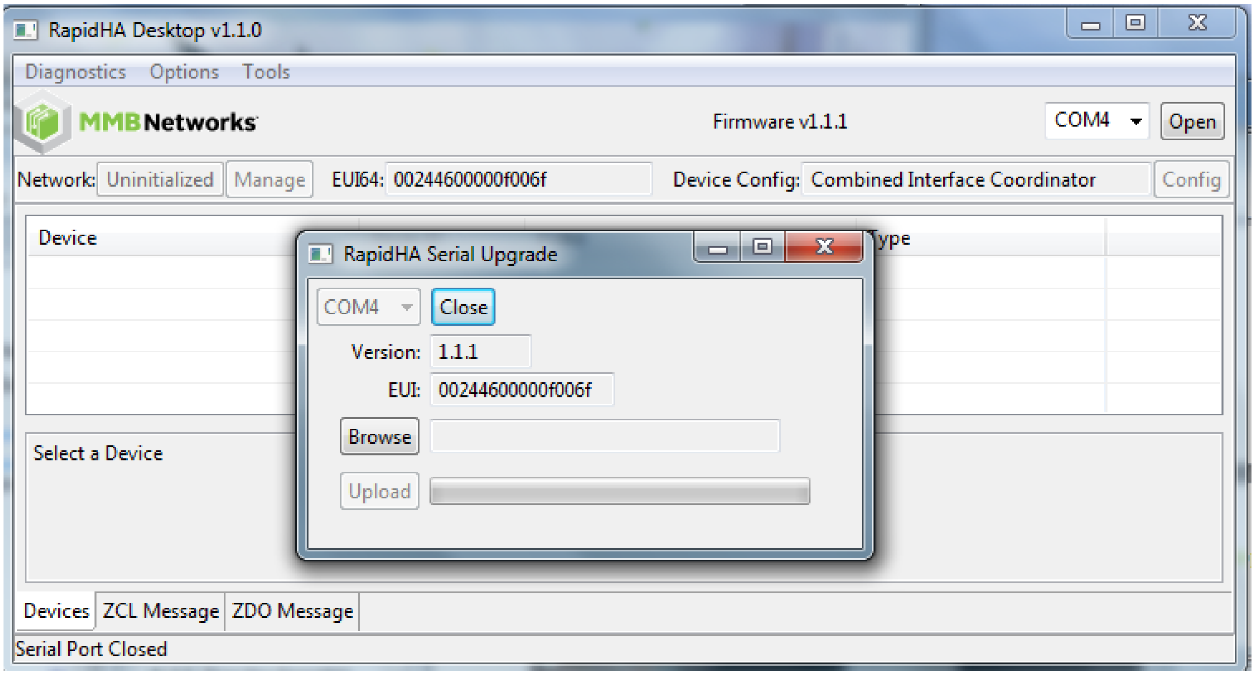
Select a Device

Serial Port Closed

The serial port must be closed in the main program window before it can be opened for the serial firmware upgrade.

The firmware version and EUI number of the attached device will be reported in the popup window and the Browse button will be active as soon as the serial port is opened. Firmware files that can be updated serially have “mbl” file extensions; e.g., RapidHA_v1.2.0_rc1_prodKey.mbl.

Figure 5: Serial Upgrade Popup Window



Firmware v1.1.1

COM4

Network: Uninitialized

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

COM4

Version: 1.1.1

EUI: 00244600000f006f

Serial Port Closed

After the desired .mbl file is opened, click on the button labeled "Upload". When the upgrade has completed, a status window will report its success.

The next several sections describe how to use the desktop application to participate in a ZigBee network.

4.4 Diagnostic Tools

There are two logs available for diagnostic purposes. The screenshot in Figure 4shows where to gain access to both.

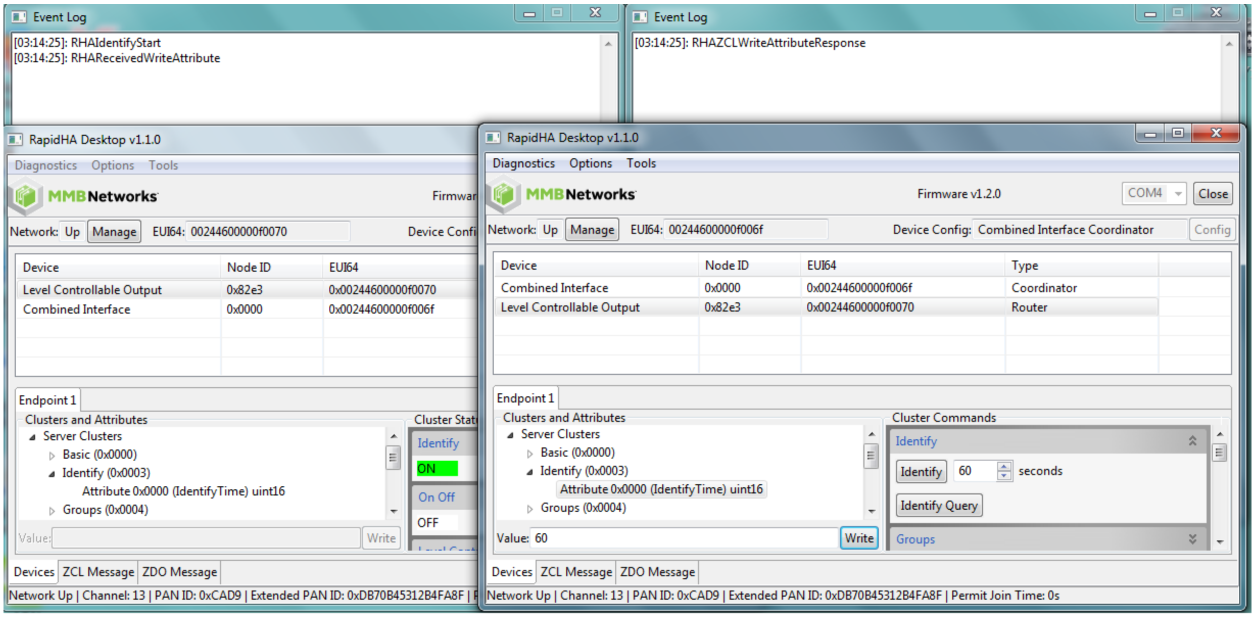
4.4.1 Event Log

Click on "Options" and select "Event Log" from the dropdown menu. The Event Log window tracks all commands sent from the Module (USB stick) to the Host (RapidHA Desktop).

Communication between network devices is also viewable in the Event Log window. The screenshot in Figure 6 shows the log results when the Identify command is sent from coordinator to end device. This was done by first selecting the end device in the coordinator's instance of RapidHA Desktop. Then the IdentifyTime attribute of the Identify cluster was selected, 60 typed in the text box and the button labeled "Write" was then clicked in order to write the value 60 seconds to the attribute.

The end device's Event Log displays the Module's response, which is to start the device's identification process by sending the Host the Identify Start command.

Figure 6: Interaction between Two Event Logs


Event Log

[03:14:25]: RHAIdentifyStart
[03:14:25]: RHAReceivedWriteAttribute

Event Log

[03:14:25]: RHAZCLWriteAttributeResponse

RapidHA Desktop v1.1.0

Network: Up | EUI64: 00244600000f0070

Device	Node ID	EUI64
Level Controllable Output	0x82e3	0x00244600000f0070
Combined Interface	0x0000	0x00244600000f006f

Network Up | Channel: 13 | PAN ID: 0xCAD9 | Extended PAN ID: 0xDB70B45312B4FA8F |

RapidHA Desktop v1.1.0

Firmware v1.2.0 | COM4

Network: Up | EUI64: 00244600000f006f | Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Level Controllable Output	0x82e3	0x00244600000f0070	Router

Value: 60

Identify 60 seconds

Network Up | Channel: 13 | PAN ID: 0xCAD9 | Extended PAN ID: 0xDB70B45312B4FA8F | Permit Join Time: 0s

4.4.2 Serial Log

Click on "Diagnostics" and select "Serial Log" from the dropdown menu. The Serial Log displays more complete command information than the Event Log. This is immediately obvious when viewing the screenshot pictured in Figure 7. The Serial Log displays the hex values of the commands sent complete with frame header, payload and checksum bytes.

Whereas the Event Log shows commands sent from the Module to the Host exclusively, the Serial Log also includes commands sent from the Host to the Module. To differentiate the contents of the Serial Log, commands sent from the Host are in bold text, while the commands sent from the Module are displayed in normal text.

The information displayed in Figure 7 is the result of opening the serial port. We will look at the first four lines in the Serial Log to explain how to decipher the information contained there.

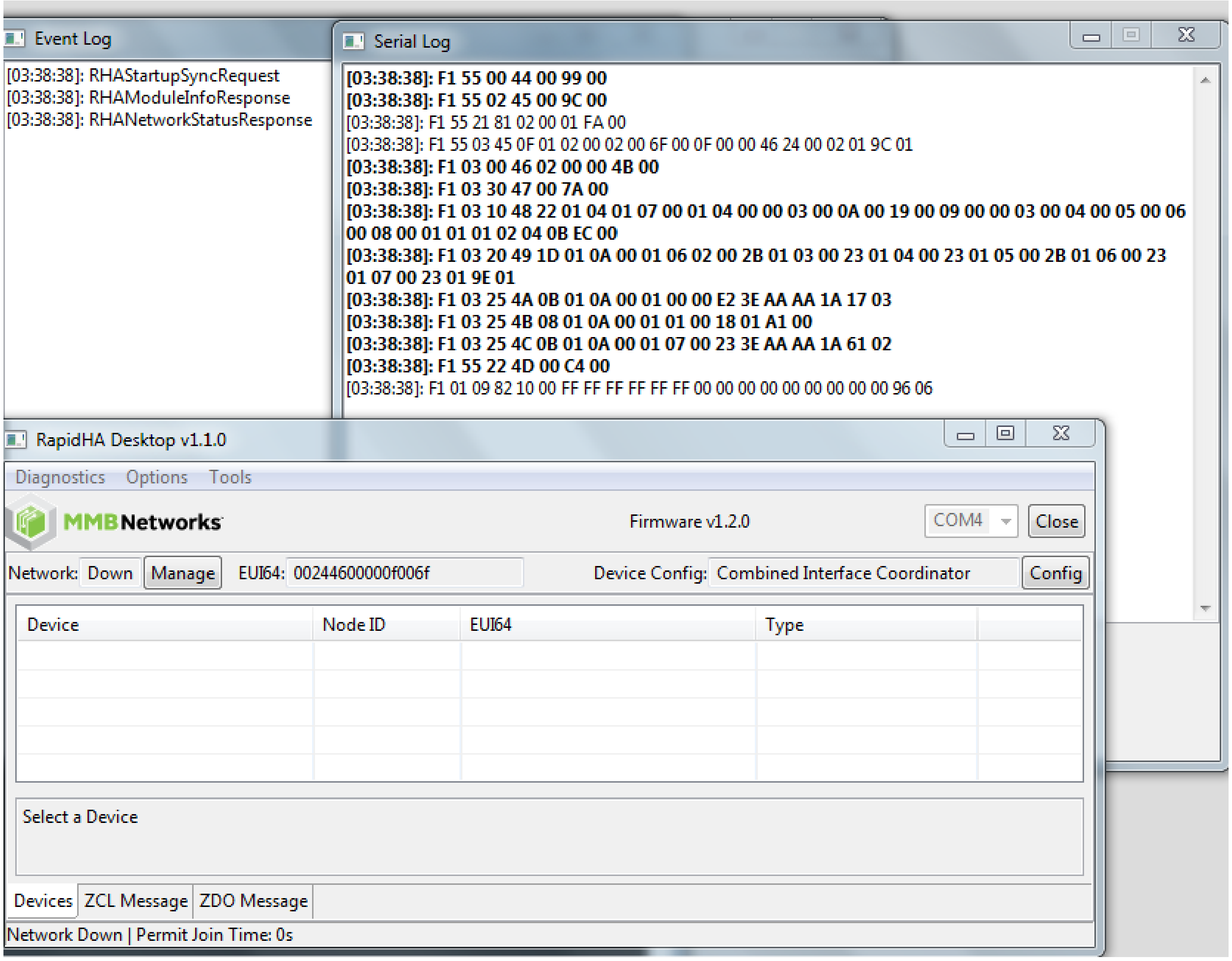
[03:38:38]: F1 55 00 44 00 99 00
[03:38:38]: F1 55 02 45 00 9C 00
[03:38:38]: F1 55 21 81 02 00 01 FA 00
[03:38:38]: F1 55 03 45 0F 01 02 00 02 00 6F 00 0F 00 00 46 24 00 02 01 9C 01

The bracketed numbers are the time in hours, minutes and seconds. Next comes the 5 bytes of the frame header, the 2nd and 3rd bytes giving us the primary and secondary headers that define the commands being sent. From this information we know the following commands were sent:

Reset (55 00)
Module Info Request (55 02)
Startup Synch Request (55 21)
Module Info Response (55 03)

Complete information on these and all other RapidHA commands can be found in the *RapidHA Serial Protocol Specification and Reference Guide*.

Figure 7: Log Information after Opening Serial Port



Event Log

```
[03:38:38]: RHAStartupSyncRequest
[03:38:38]: RHAModuleInfoResponse
[03:38:38]: RHANetworkStatusResponse
```

Serial Log

```
[03:38:38]: F1 55 00 44 00 99 00
[03:38:38]: F1 55 02 45 00 9C 00
[03:38:38]: F1 55 21 81 02 00 01 FA 00
[03:38:38]: F1 55 03 45 0F 01 02 00 02 00 6F 00 0F 00 00 46 24 00 02 01 9C 01
[03:38:38]: F1 03 00 46 02 00 00 4B 00
[03:38:38]: F1 03 30 47 00 7A 00
[03:38:38]: F1 03 10 48 22 01 04 01 07 00 01 04 00 00 03 00 0A 00 19 00 09 00 00 03 00 04 00 05 00 06
00 08 00 01 01 01 02 04 0B EC 00
[03:38:38]: F1 03 20 49 1D 01 0A 00 01 06 02 00 2B 01 03 00 23 01 04 00 23 01 05 00 2B 01 06 00 23
01 07 00 23 01 9E 01
[03:38:38]: F1 03 25 4A 0B 01 0A 00 01 00 00 E2 3E AA AA 1A 17 03
[03:38:38]: F1 03 25 4B 08 01 0A 00 01 01 00 18 01 A1 00
[03:38:38]: F1 03 25 4C 0B 01 0A 00 01 07 00 23 3E AA AA 1A 61 02
[03:38:38]: F1 55 22 4D 00 C4 00
[03:38:38]: F1 01 09 82 10 00 FF FF FF FF FF FF 00 00 00 00 00 00 00 00 00 96 06
```

Firmware v1.2.0

COM4

Network: Down

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type

Select a Device

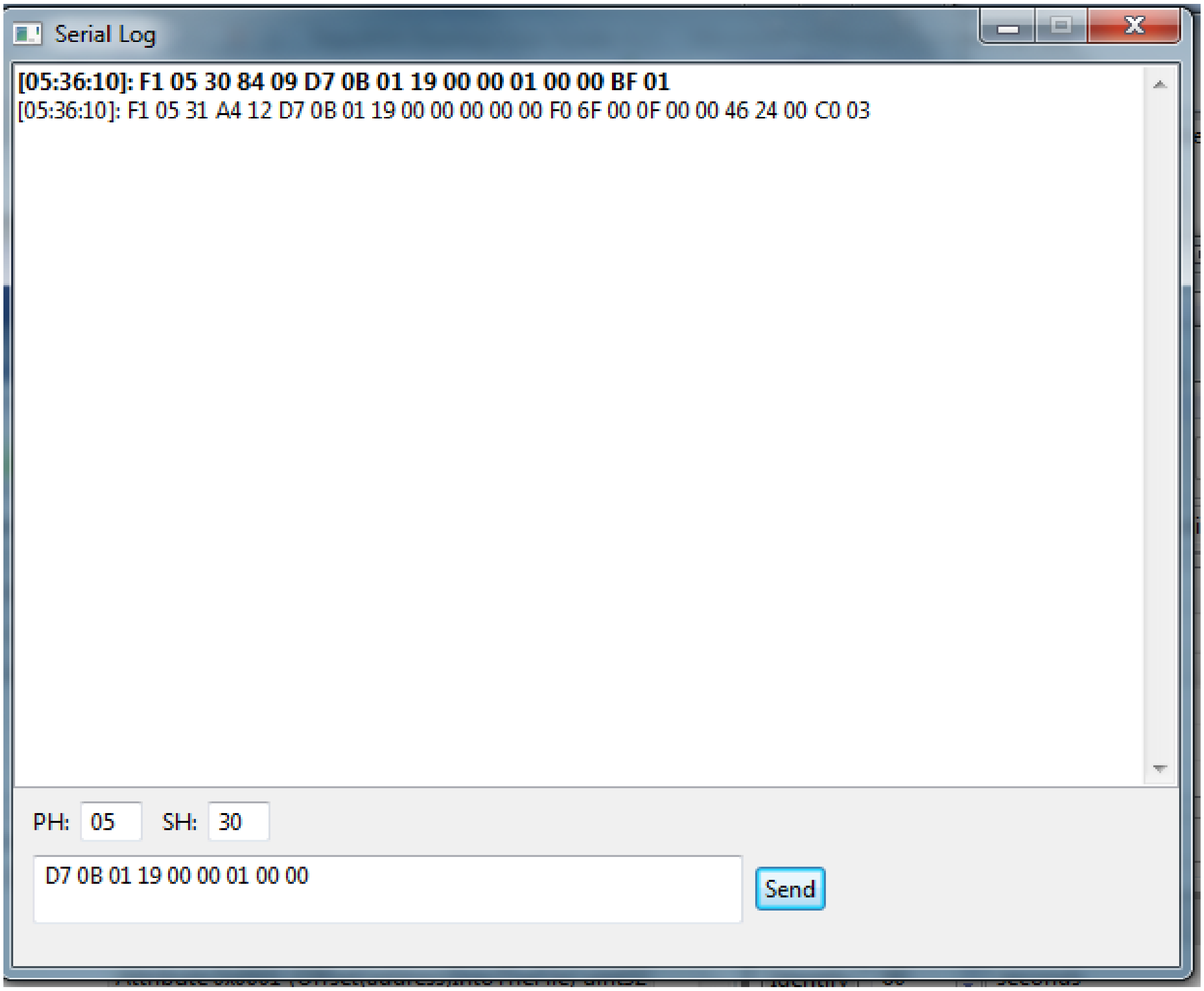
Network Down | Permit Join Time: 0s

The Serial Log can also be used to send commands, as is shown in Figure 8. At the bottom of the window are three textboxes that contain the primary and secondary headers and the command payload if any. In our example these values are:

PH = Primary Header = 0x05
SH = Secondary Header = 0x30
Payload = D7 0B 01 19 00 00 01 00 00

This translates to the ZCL Read Attribute Request, specifically a request to read the value of attribute 0x0000 (OtaUpgradeServerId) from cluster ID 0x0019 (OTA Upgrade cluster). The Module responds with the IEEE address of the coordinator.

Figure 8: Sending a Command via the Serial Log Window



```
[05:36:10]: F1 05 30 84 09 D7 0B 01 19 00 00 01 00 00 BF 01
[05:36:10]: F1 05 31 A4 12 D7 0B 01 19 00 00 00 00 00 F0 6F 00 0F 00 00 46 24 00 C0 03
```

PH: 05 SH: 30

D7 0B 01 19 00 00 01 00 00

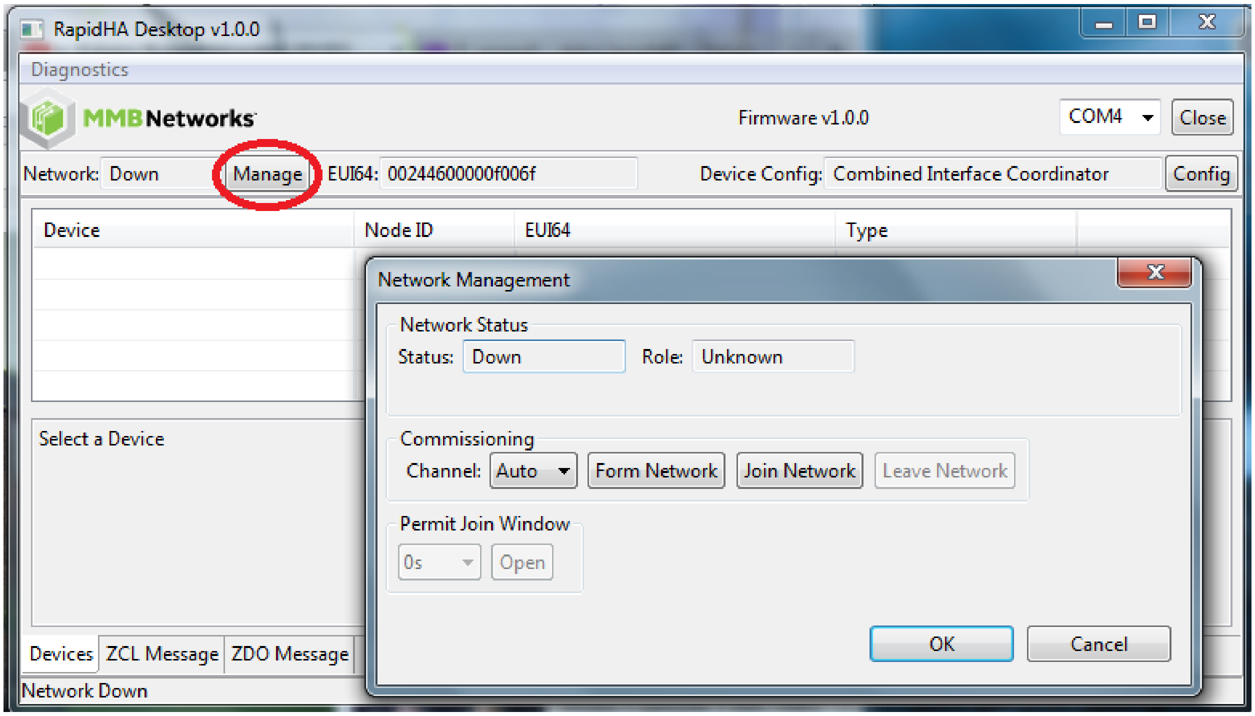
5.0 Network Management

To perform network tasks, click on the button labeled “Manage” located on the upper left side of the program window. This button is pictured in Figure 9, along with the Network Management popup window that it brings up.

From the Network Management popup window, the following tasks may be performed:

- Form a network and act as the coordinator
- Join a network as a router or end device
- Leave a network (when the coordinator leaves a network, the network no longer exists)
- Permit other devices to join a network

Figure 9: Manage the Network



RapidHA Desktop v1.0.0

Firmware v1.0.0

COM4

Network: Down

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type

Select a Device

Network Down

Network Management

Network Status

Status: Down

Role: Unknown

Commissioning

Channel: Auto

Permit Join Window

0s

5.1 Network Status

The network status is reported both at the top and the bottom of the program window. Opening the serial port will result in one of the following network states:

- **Network: Uninitialized** – this state is reached when the desktop application executes for the first time.
- **Network: Down** – this state is reached when the device leaves the network. If the coordinator leaves, the network no longer exists.
- **Network: Up** – this state is reached when a coordinator forms a network or when another device joins that network. The channel, PAN ID and extended PAN ID of the network will be reported in the status line at the bottom of the program window. If the Network Management popup window is open, the same information will also be reported there. These parameters identify a unique ZigBee network. See Figure 10 for an example.

5.2 Network Commissioning Options

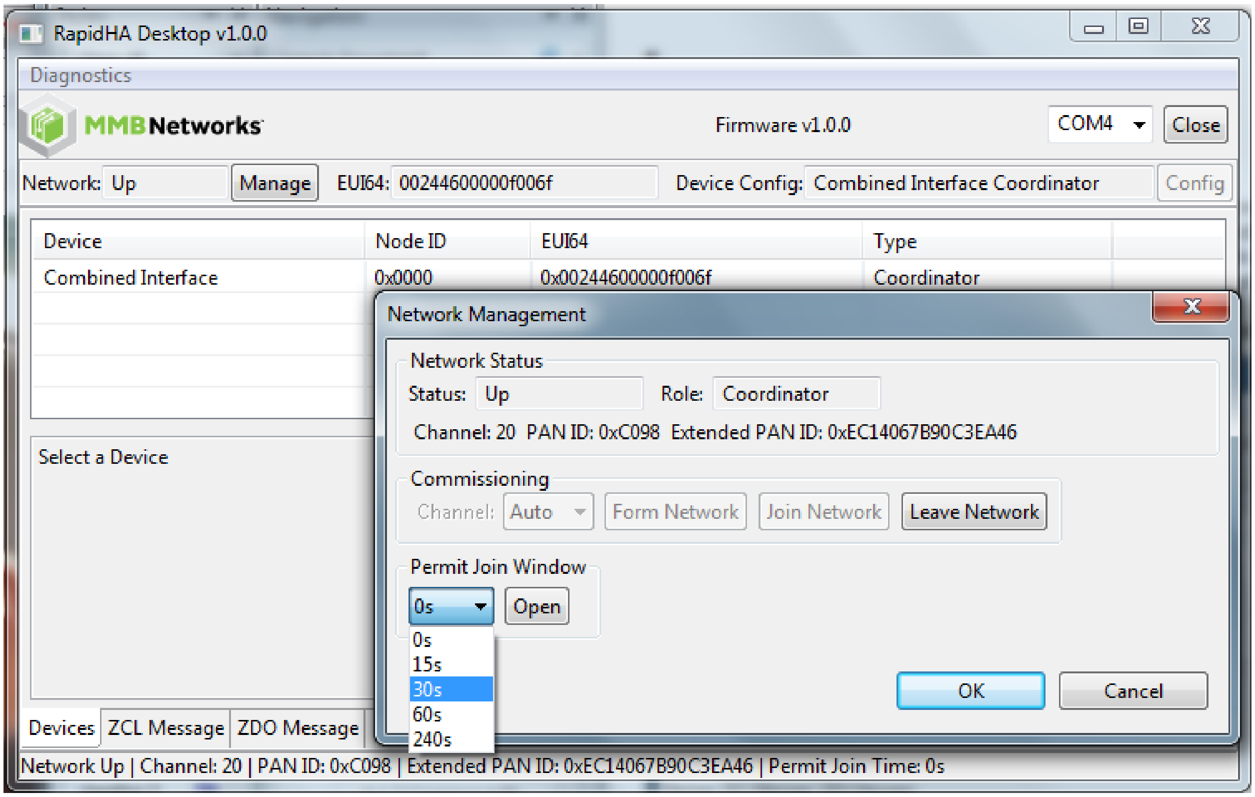
The network state determines which buttons are active in the Commissioning area of the Network Management popup window.

If the network is down, as illustrated in Figure 9, the device may form or join a network. The ZigBee logical device type is tied to this action. If the Form Network button is clicked and the action is successful, the device takes on the role of coordinator. If the Join Network button is clicked, the device becomes a router.

Before joining or forming a network, a channel may be chosen from the dropdown menu. Selecting a specific channel limits the device to joining or forming a network on that channel only. The default value of “Auto” allows the ZigBee coordinator to select the channel with the least interference when it forms a network and it allows a joining device to join any available network no matter which channel it operated on.

If the network is up, as illustrated in Figure 10, you can allow other devices to join the network (see Section 5.3 for more information). You can leave the network by clicking on the Leave Network button.

Figure 10: Network Up and Permit Join Window Available



Firmware v1.0.0

Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator

Select a Device

Status: Up Role: Coordinator

Channel: 20 PAN ID: 0xC098 Extended PAN ID: 0xEC14067B90C3EA46

Channel: Auto

Permit Join Window: 0s

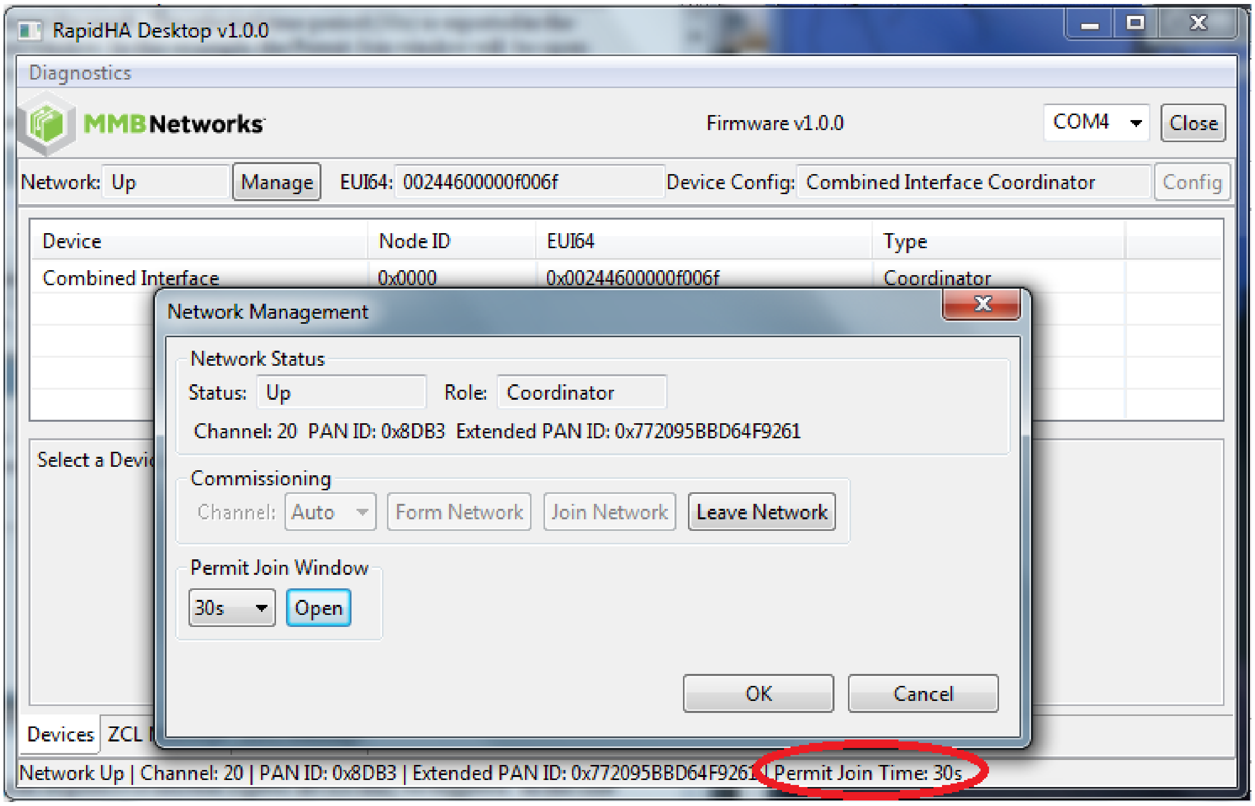
Network Up | Channel: 20 | PAN ID: 0xC098 | Extended PAN ID: 0xEC14067B90C3EA46 | Permit Join Time: 0s

5.3 Permit Join Window

After the network has formed, the coordinator must open the Permit Join Window in order to let other devices join the network.

From the dropdown menu, select one of the four time periods available and then click the button labeled “Open”. The screenshot in Figure 11 shows the result. The selected time period (30s) is reported in the status line at the bottom of the program window. In this example, the Permit Join Window will be open until the system counts down thirty seconds.

Figure 11: Permit Join Window is Open



Firmware v1.0.0

COM4

Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator

Network Status

Status: Up Role: Coordinator

Channel: 20 PAN ID: 0x8DB3 Extended PAN ID: 0x772095BBD64F9261

Commissioning

Channel: Auto

Permit Join Window

30s

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 30s

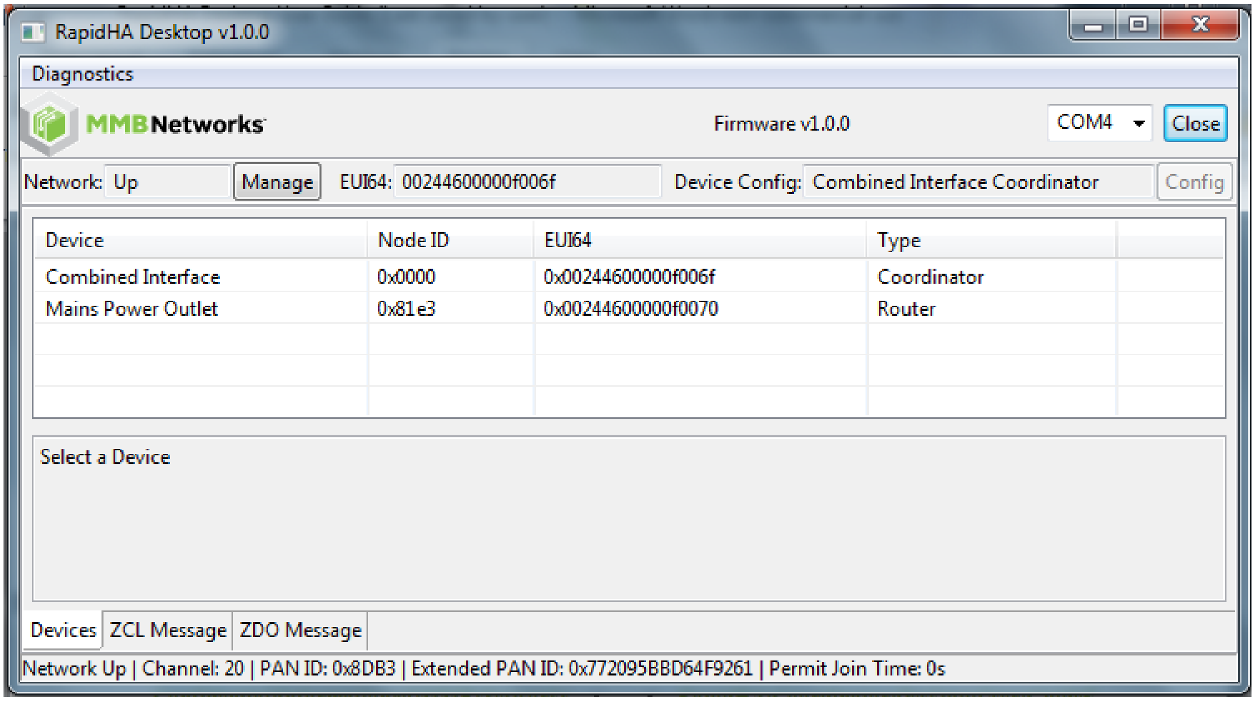
Once the Open button is clicked, the join sequence of another ZigBee device may be triggered.

In the case of a second RapidConnect USB stick, the join sequence involves opening the Network Management popup window of the second instance of the desktop program. From this popup window, click on the button labeled “Join Network”. When the join process completes, both instances of the program will reflect the updated membership of the network.

Figure 12 shows the program window associated with the coordinator and Figure 13 shows the program window associated with the device that joined the network as a router.

The USB sticks can be physically identified by their EUI64 numbers. This is the Extended Universal Identifier printed on a label affixed to the back of the device. This number will match the EUI64 number reported in the program window.

Figure 12: Program Window of Coordinator



Firmware v1.0.0

Network: Up

EUI64: 00244600000f006f

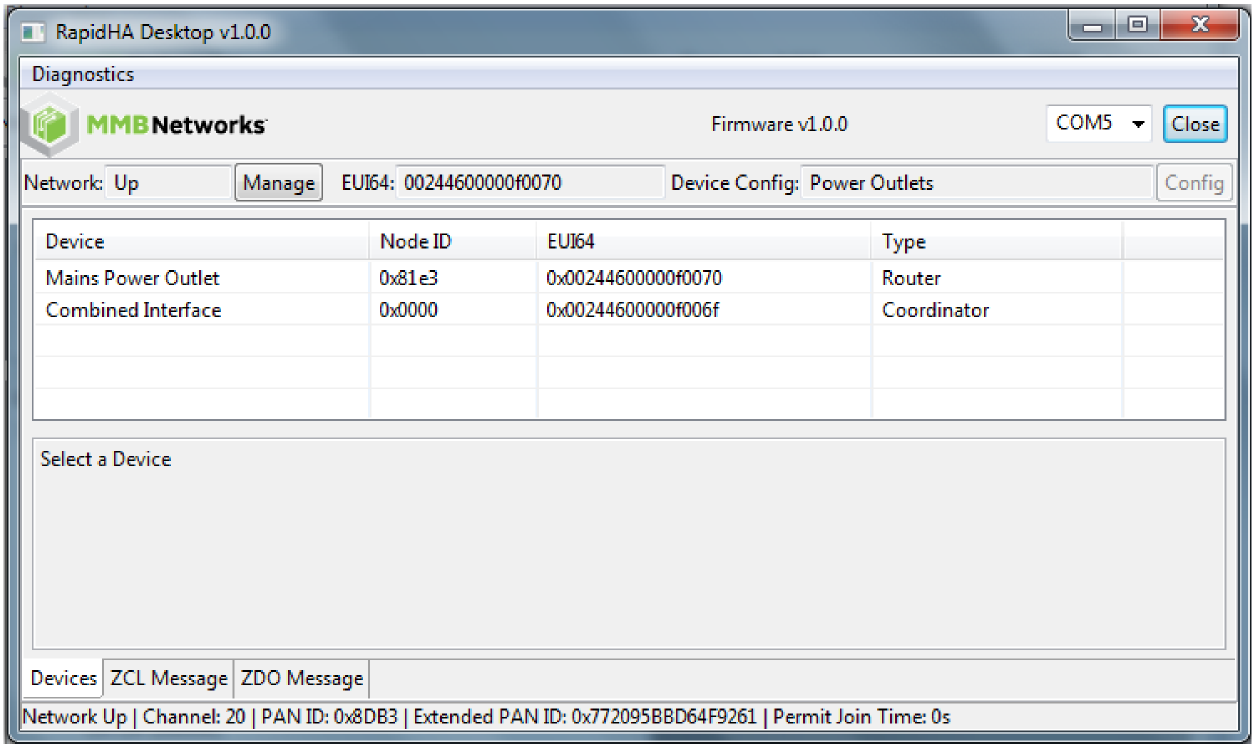
Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Mains Power Outlet	0x81e3	0x00244600000f0070	Router

Select a Device

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

Figure 13: Program Window of Router



Firmware v1.0.0

Network: Up

EUI64: 00244600000f0070

Device Config: Power Outlets

Device	Node ID	EUI64	Type
Mains Power Outlet	0x81e3	0x00244600000f0070	Router
Combined Interface	0x0000	0x00244600000f006f	Coordinator

Select a Device

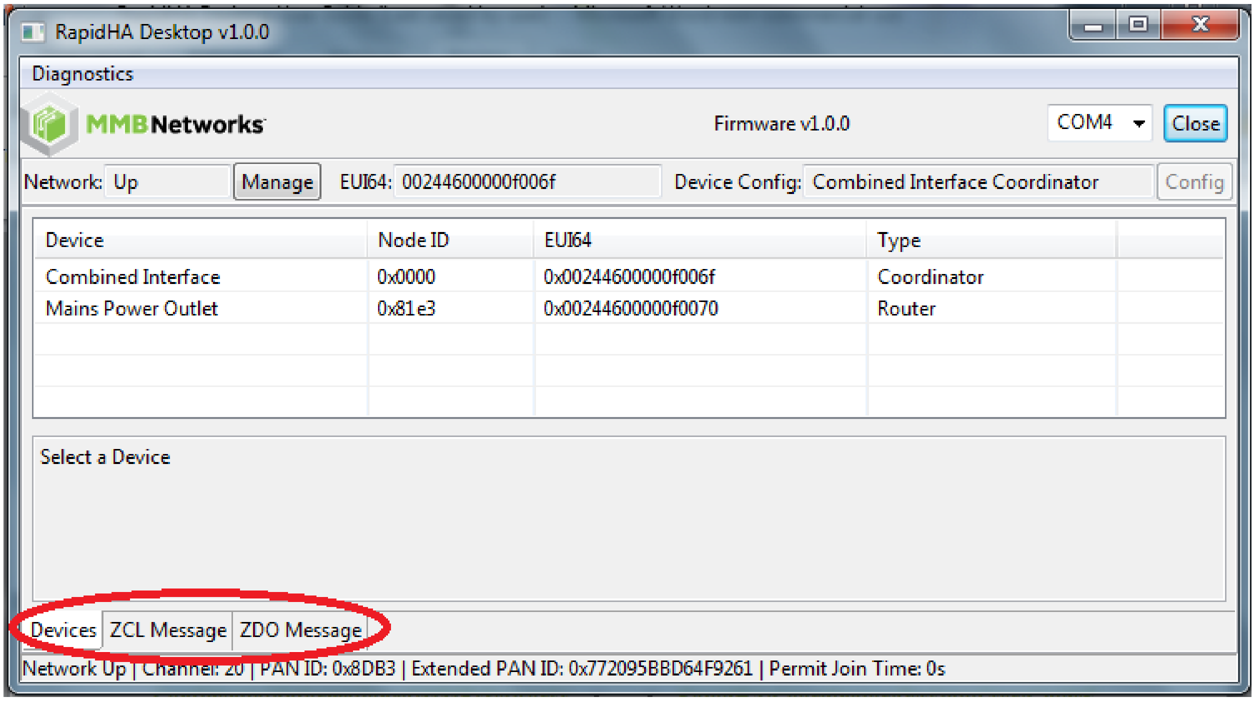
Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

6.0 Devices Tab

As previously described, at the top and bottom of the program window there are status lines that report on network connectivity, along with buttons that facilitate that connectivity. The content of the middle part of the program window is determined by one of three tabs (see Figure 14 for tab locations).

By default, RapidHA Desktop always begins execution with the “Devices” tab active. On this tab, each node in the network is described and identified with a line in the Devices area. The example in Figure 14 shows two nodes, 0x0000 (always the node ID of the coordinator) and 0x81e3.

Figure 14: Devices Tab Screenshot



Firmware v1.0.0

Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Mains Power Outlet	0x81e3	0x00244600000f0070	Router

Select a Device

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

6.1 Select a Device

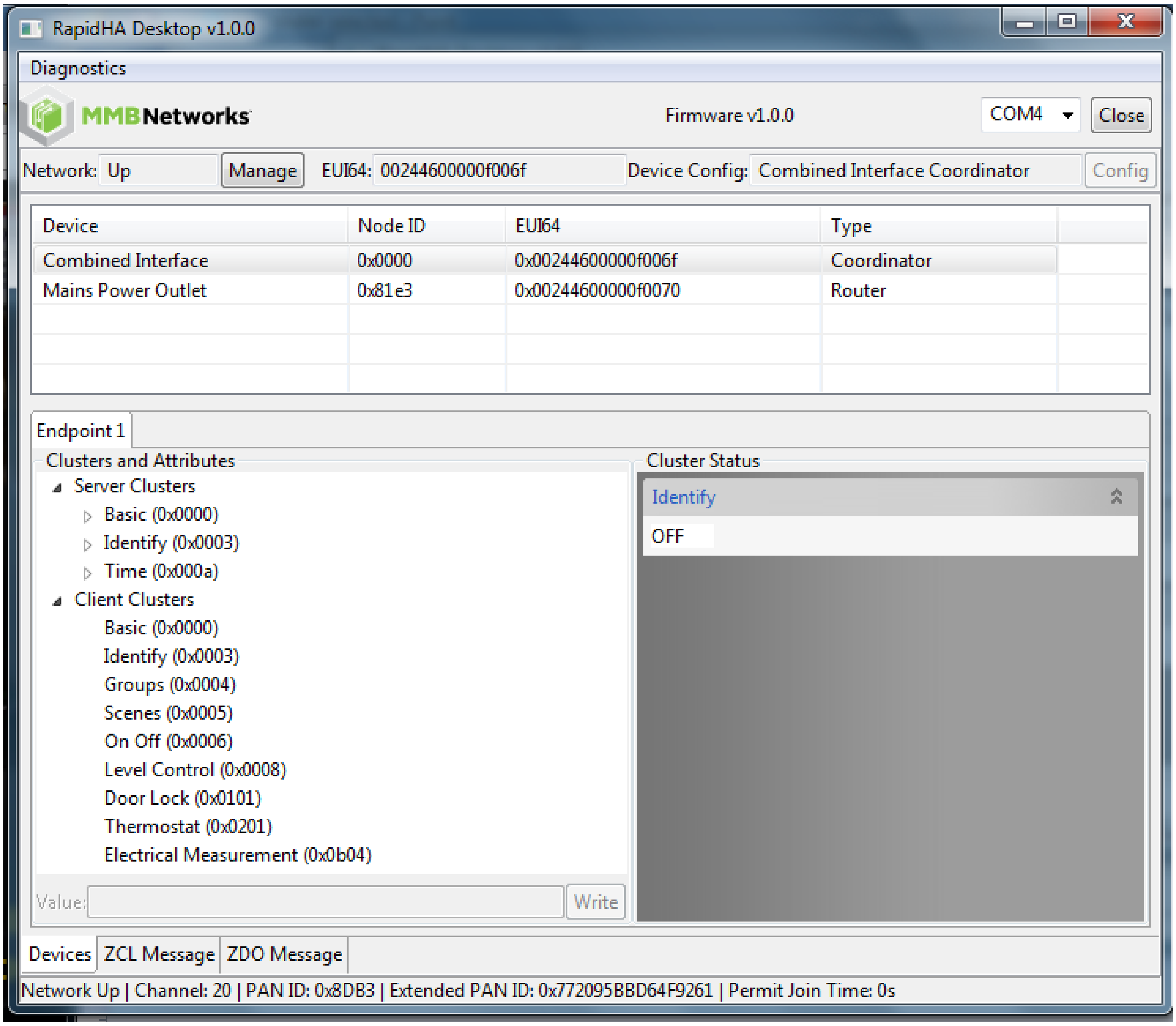
Click on a line in the Devices area to populate the lower half of the window with the endpoints, clusters and attributes defined on that device. The screenshot in Figure 15 shows the coordinator-associated instance of RapidHA Desktop with the coordinator selected in the Devices area.

An important thing to note is that the coordinator is configured with client clusters. This allows for the sending of commands to the corresponding server clusters via the cluster command interface on the Devices tab.

The command interface appears to the right of the endpoint data. This area of the window is for “Cluster Status” or “Cluster Commands”. What appears there, status or commands, depends on which device is selected and which program instance is associated with the device. For example, in Figure 15, the coordinator-associated program instance is pictured and the coordinator is selected; thus, the status of the coordinator’s Identity cluster appears.

To see the cluster commands available in this area for the coordinator to send to node 0x81e3, that node would have to be selected in the Devices area of the coordinator’s program instance. This is shown in the screenshot in Figure 16.

Figure 15: Device Selected



Firmware v1.0.0

Network: Up | EUI64: 00244600000f006f | Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Mains Power Outlet	0x81e3	0x00244600000f0070	Router

- Server Clusters
  - Basic (0x0000)
  - Identify (0x0003)
  - Time (0x000a)
- Client Clusters
  - Basic (0x0000)
  - Identify (0x0003)
  - Groups (0x0004)
  - Scenes (0x0005)
  - On Off (0x0006)
  - Level Control (0x0008)
  - Door Lock (0x0101)
  - Thermostat (0x0201)
  - Electrical Measurement (0x0b04)

Value:

Identify

OFF

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

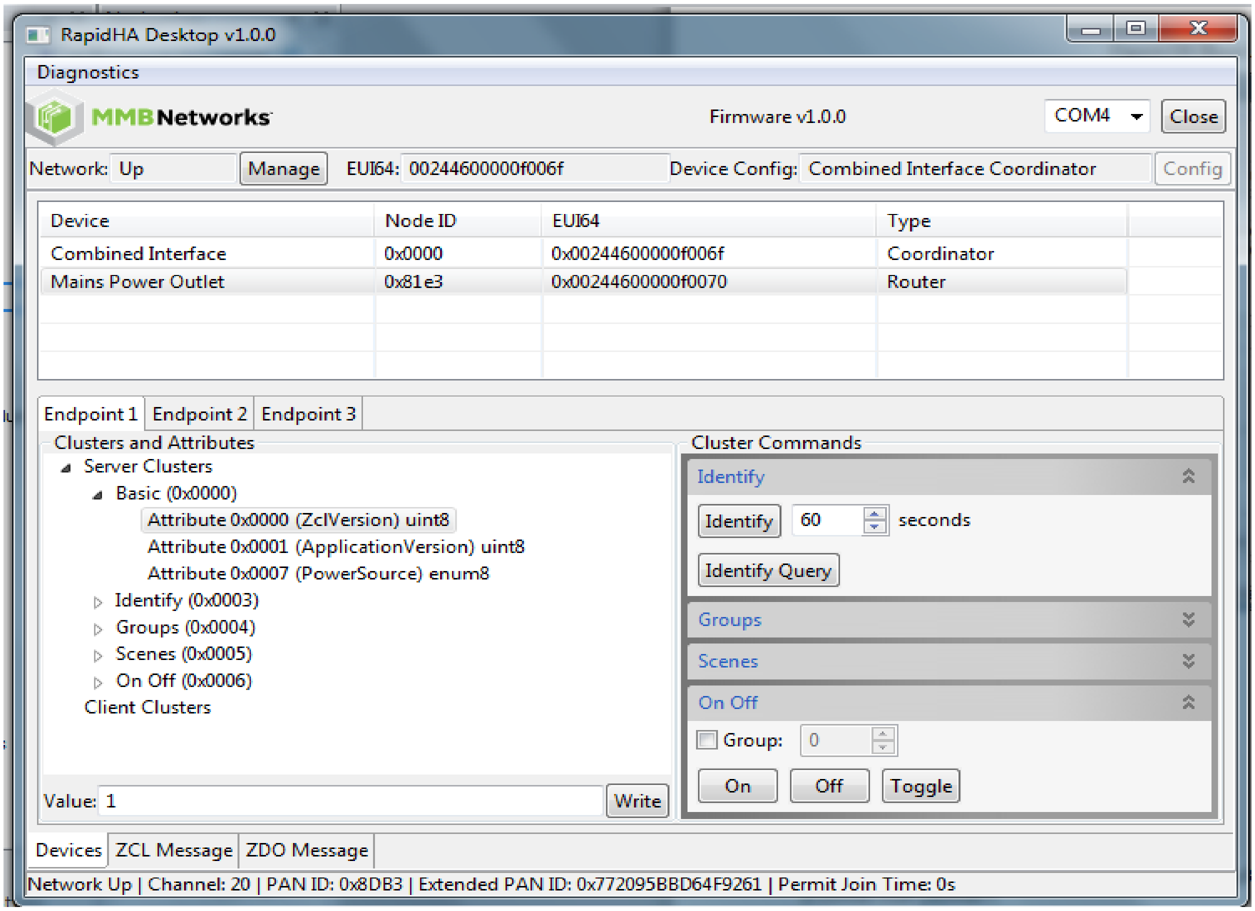
6.2 Read/Write Attributes

To read or write an attribute from the Devices tab, select the attribute as shown in Figure 16, where an attribute from the Basic cluster is highlighted, i.e., selected. Notice that the value of the selected attribute is “1” as reported in the textbox labeled “Value”. To write to the attribute, type a new value into the textbox and click on the button to the right labeled “Write”.

Follow these rules for writing values to attributes:

- Unsigned integers, signed integers, and enumerations can be written with a decimal value; e.g., 60, 100, 0
- Boolean types must be written as “false” and “true”; these keywords are case-sensitive
- All other data types must be written as hexadecimal byte arrays, with least significant byte first; e.g., 01 02 03 04

Figure 16: Read Attribute



Firmware v1.0.0 COM4

Network: Up EUI64: 00244600000f006f Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Mains Power Outlet	0x81e3	0x00244600000f0070	Router

Clusters and Attributes

- Server Clusters
  - Basic (0x0000)
    - Attribute 0x0000 (ZclVersion) uint8
    - Attribute 0x0001 (ApplicationVersion) uint8
    - Attribute 0x0007 (PowerSource) enum8
  - Identify (0x0003)
  - Groups (0x0004)
  - Scenes (0x0005)
  - On Off (0x0006)
- Client Clusters

Value: 1

Cluster Commands

Identify 60 seconds

□ Group: 0

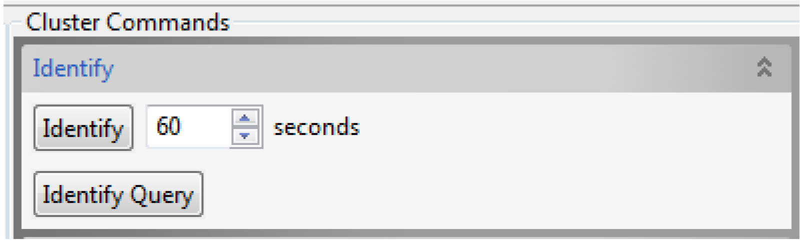
Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

6.3 Cluster Commands

The cluster commands that can be sent from the Devices tab depend on which ZigBee device is selected and which server clusters are available on the endpoint whose tab is active in the program window. Note that the screenshot in Figure 16 shows Endpoint 1 as active.

6.3.1 Identify Commands

The Identify cluster provides a way to visually identify a device to an observer, e.g., a flashing light. The RapidHA desktop uses the Cluster Status area of the program window to simulate a device identifying itself (see Figure 27). The Identify command can be sent from the Cluster Commands interface or the IdentifyTime attribute may be written directly.

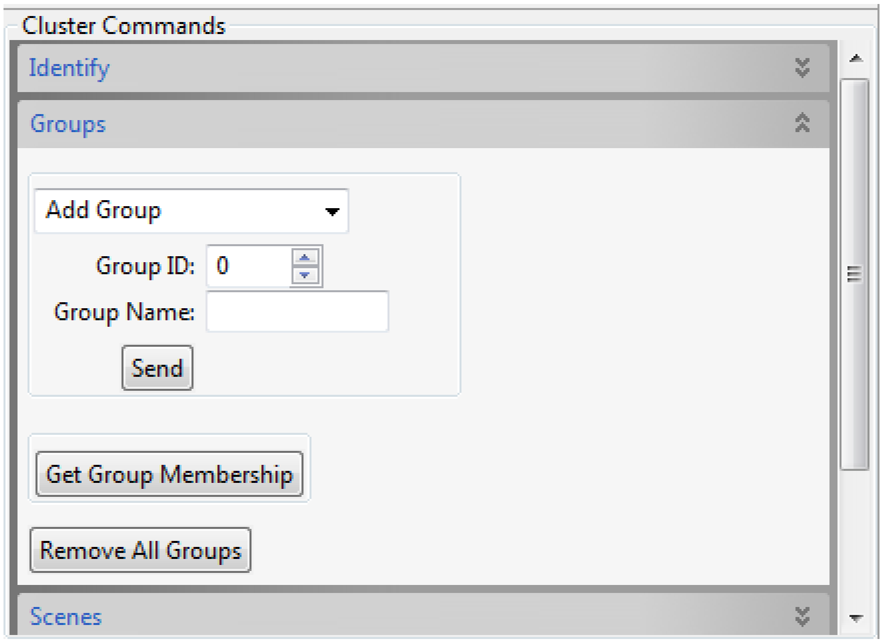


Identify Query requests all devices that are identifying themselves to respond to the sender of the command.

6.3.2 Group Commands

The Group cluster allows for group addressing. Commands defined for the Group cluster can be sent from the command interface on the Devices tab. The first four commands are accessible from the dropdown menu.

Figure 17: Group Cluster Commands



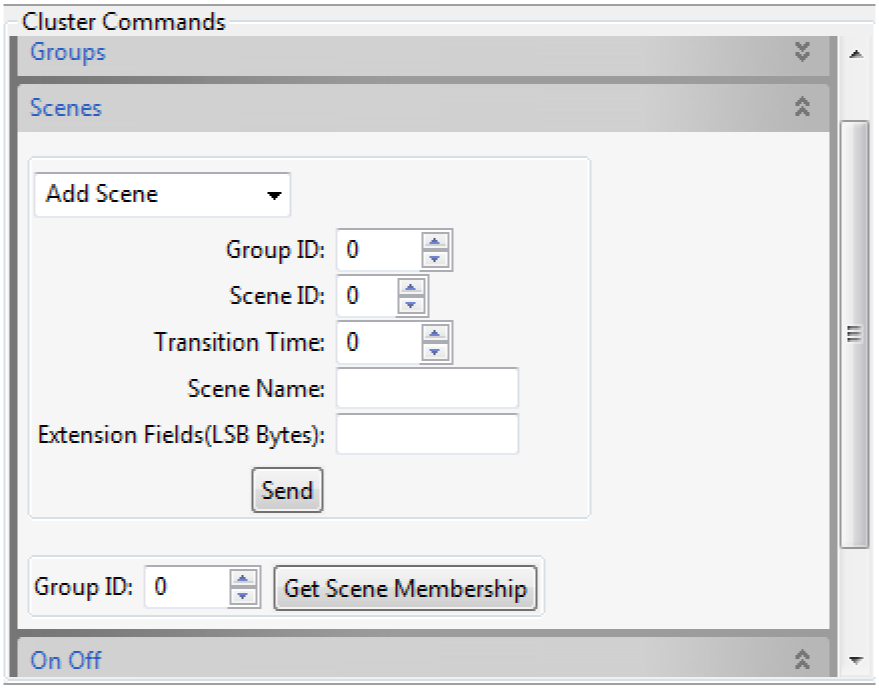
1. **Add Group** – adds the destination endpoint’s membership to the specified group. The group will be created if it does not already exist.
2. **Add Group if Identifying** – add group membership to an endpoint on the receiving device if it is identifying itself; i.e., the IdentifyTime attribute on the destination endpoint is non-zero.
3. **View Group** – get the Group Name string for the specified group. If names are supported, it will display as ASCII in the “Group Name” textbox.
4. **Remove Group** – remove the destination endpoint’s membership from the specified group.
5. **Remove All Groups** – remove all group associations on the destination endpoint.

The Group Name is optional. If names are supported, the most significant bit is set of the Group cluster’s attribute, NameSupport.

6.3.3 Scene Commands

A scene is a collection of stored attribute values. The Scene cluster contains attributes and clusters to set up and recall scenes. From the command interface one can add, view, remove, store, and recall specific scenes. These commands are selected from the dropdown menu and are then sent by clicking the button labeled “Send”.

Figure 18: Scene Cluster Commands



Add Scene

Group ID: 0

Scene ID: 0

Transition Time: 0

Scene Name:

Extension Fields(LSB Bytes):

Group ID: 0

- **Group ID**: a scene is typically associated with a group; if it is not, then the Group ID must equal zero.
- **Scene ID**: user-selected identifier; range 0 to 255, inclusive.
- **Transition Time**: the number of seconds it takes for the requested scene to replace the current state of the affected attributes.
- **Scene Name**: this string is optional. If names are supported, the most significant bit is set of the Scene cluster’s attribute, NameSupport.
- **Extension Fields** (LSB Bytes): a list of one or more clusters and the state of their associated attributes which will be part of the scene being created. To be part of a scene, a cluster must have an attribute named in a Scene Table Extension section in the *ZigBee Cluster Library Specification* or the *Home Automation Public Profile*.

  There are three parts for each entry:

  1. ClusterID: 2-byte identifier for the cluster of interest.
  2. Length: number of bytes for the extension field set(s) that follow.
  3. Extension Field Set: the attribute value(s) associated with the cluster.

For example the entry “06 00 01 01” means:

1. ClusterID: 06 00 = 0x0006 = On/Off cluster
2. Length: 01 = the following extension field set is one byte in length
3. Extension Field Set: 01 = set the OnOff attribute to “On”

To include another cluster, simply append it. For example, entering “06 00 01 01 08 00 01 FF” would include the Level Control cluster to the scene, assigning 0xFF to the attribute CurrentLevel whenever the scene is recalled.

6.3.4 On/Off Commands

The commands available for this cluster are “On”, “Off”, and “Toggle”. There is also an option to check “Group” which allows you to send the desired command only to the endpoints associated with the Group ID specified.

6.4 Time Cluster

The Time cluster provides an interface to a real-time clock.

6.4.1 Server-Side

The server-side of the Time cluster provides a set of commands and correlating attributes. The GUI is pictured in Figure 19.

6.4.1.1 Commands

Server-side cluster commands are described here. They are grouped into three basic areas.

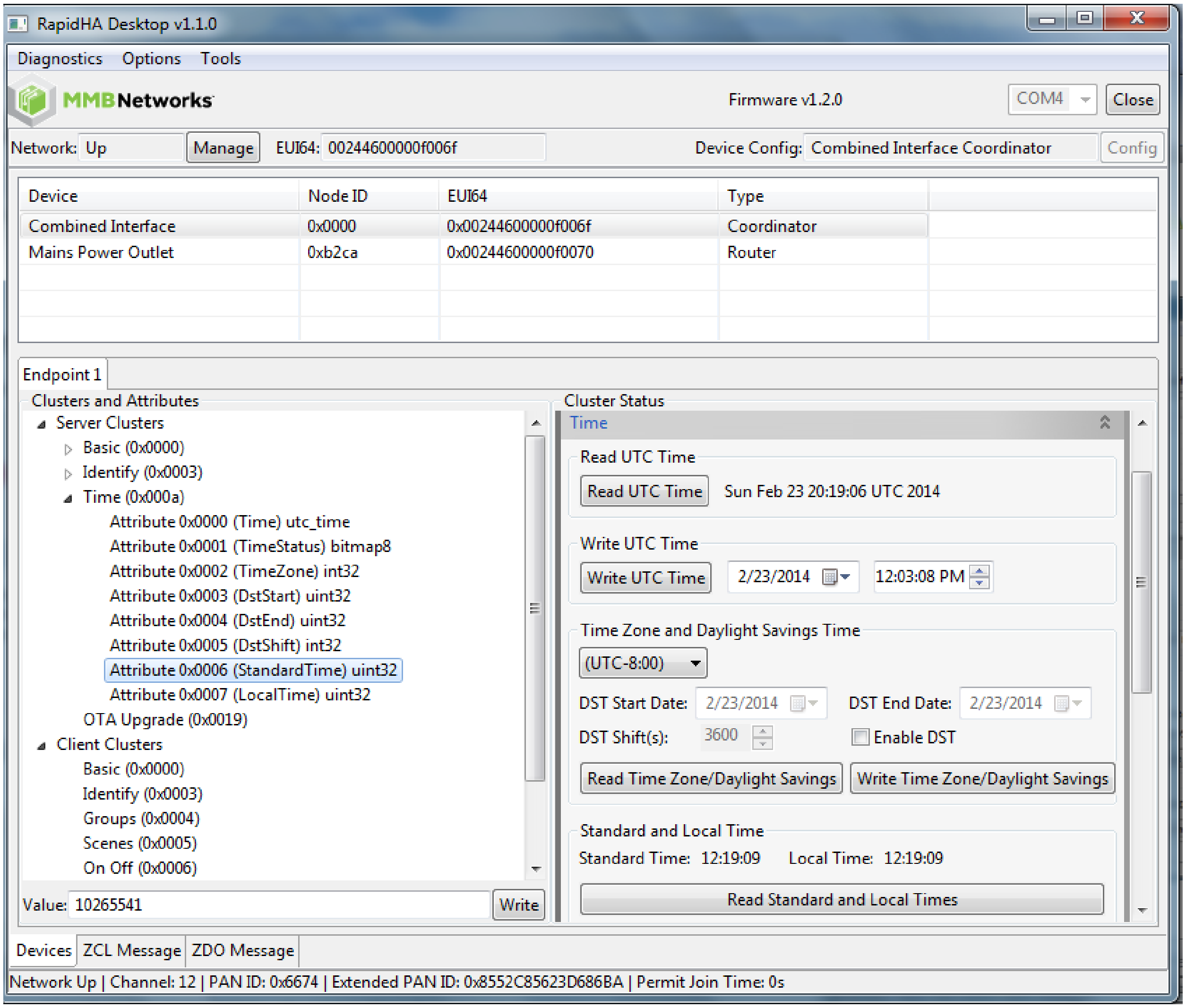
1. **Read/Write UTC Time** – The time value used by the RapidHA Desktop synchronizes with the real-time clock of the PC on which the program is running. Click on the button labeled “Read UTC Time” to update the textbox with the current value of the real-time clock.

   You may decouple the association between the real-time clock and the time used by RapidHA Desktop. To do so, click on the button labeled “Write UTC Time” after the desired date and time have been specified in their respective textboxes to the right of the button.

   Click on “Read UTC Time” to see the value just written. Being able to write the time value allows more flexibility; e.g., to record the number of seconds since an event such as device initialization.

   Please note that changing the time value used by RapidHA Desktop has no effect on the real-time clock. If the USB stick is reset, the RapidHA Desktop will again synchronize with the PC’s real-time clock. A reset occurs when the serial port is closed and also with the “Reset Module” command, an option available on the Diagnostics dropdown menu located in the upper left side of the program window.

Figure 19: Time Cluster, Server-Side



Firmware v1.2.0

Network: Up EUI64: 00244600000f006f Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator
Mains Power Outlet	0xb2ca	0x00244600000f0070	Router

Clusters and Attributes

- Server Clusters
  - Basic (0x0000)
  - Identify (0x0003)
  - Time (0x000a)
    - Attribute 0x0000 (Time) utc_time
    - Attribute 0x0001 (TimeStatus) bitmap8
    - Attribute 0x0002 (TimeZone) int32
    - Attribute 0x0003 (DstStart) uint32
    - Attribute 0x0004 (DstEnd) uint32
    - Attribute 0x0005 (DstShift) int32
    - Attribute 0x0006 (StandardTime) uint32
    - Attribute 0x0007 (LocalTime) uint32
  - OTA Upgrade (0x0019)
- Client Clusters
  - Basic (0x0000)
  - Identify (0x0003)
  - Groups (0x0004)
  - Scenes (0x0005)
  - On Off (0x0006)

Value: 10265541

Cluster Status

Time

Read UTC Time

Sun Feb 23 20:19:06 UTC 2014

Write UTC Time

2/23/2014 12:03:08 PM

Time Zone and Daylight Savings Time

(UTC-8:00)

DST Start Date: 2/23/2014 DST End Date: 2/23/2014

DST Shift(s): 3600 □ Enable DST

Standard and Local Time

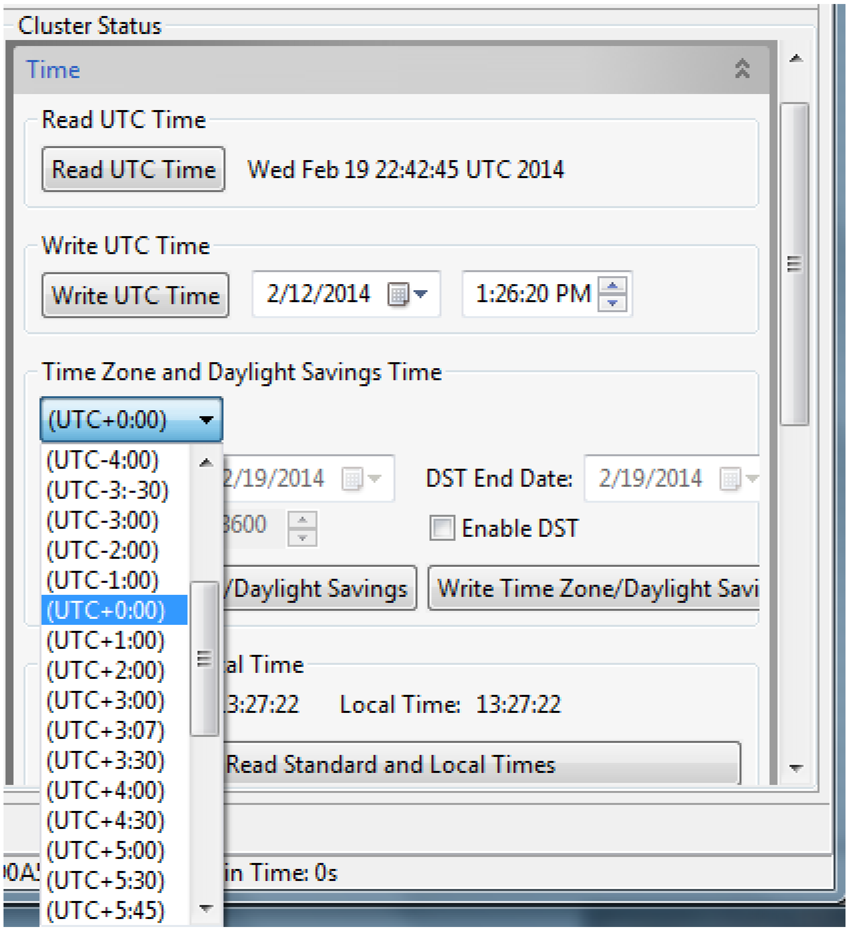
Standard Time: 12:19:09 Local Time: 12:19:09

Network Up | Channel: 12 | PAN ID: 0x6674 | Extended PAN ID: 0x8552C85623D686BA | Permit Join Time: 0s

2. **Time Zone/ Daylight Saving Time** – The time zone is an offset from the UTC standard. This information is location-specific. It is easy to determine time zone and daylight saving time dates for any location. There are various websites available for this purpose.

   The time zone, once it has been determined, is selected from the dropdown menu pictured in Figure 20. Many geographical locations also use daylight saving time. A checkbox labeled "Enable DST" when checked allows start and end date values for daylight saving time to be entered. The value for "DST Shift(s)" is the number of seconds added during the daylight saving time period to determine local time.

Figure 20: Time Zone Dropdown Menu



Read UTC Time: Wed Feb 19 22:42:45 UTC 2014

Write UTC Time: 2/12/2014 1:26:20 PM

Time Zone: (UTC+0:00)

2/19/2014 DST End Date: 2/19/2014

3600 □ Enable DST

3:27:22 Local Time: 13:27:22

Read Standard and Local Time – After RapidHA Desktop has synchronized with the PC's real-time clock (by clicking on the button labeled "Read UTC Time") and the Time Zone offset has been selected from the dropdown menu pictured in Figure 20, then the standard and local time can be read and displayed with a 24-hour clock. This is done by clicking on the button labeled "Read Standard and Local Times".

6.4.1.2 Attributes

The Time cluster attributes are read/write variables. This interface is pictured in Figure 19.

The available attributes are summarized here and fully described in the ZigBee Cluster Library specification:

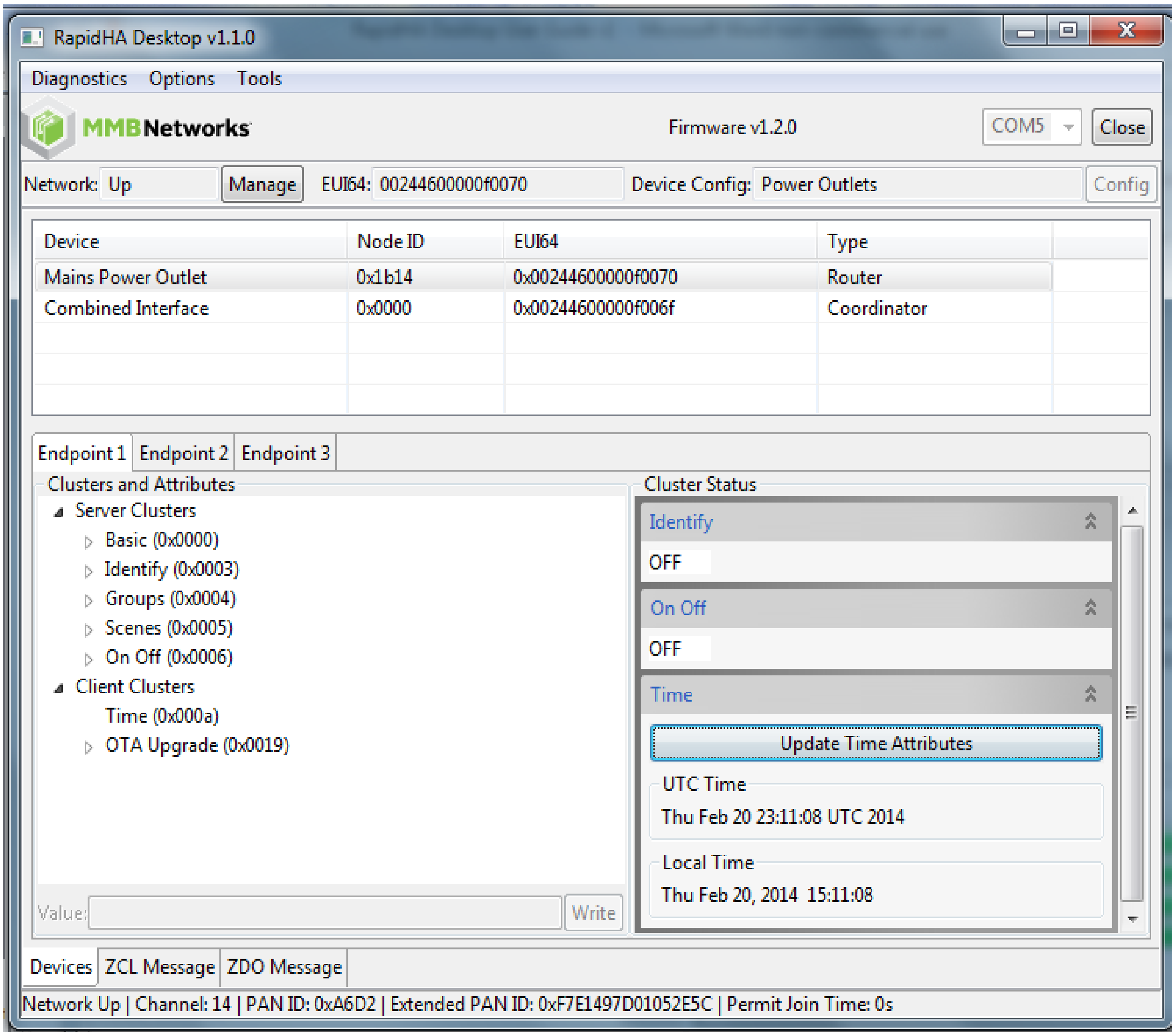
- Attribute 0x0000 (Time) – holds the value of the real-time clock or another value explicitly written either through the cluster command interface or by writing to the attribute directly. This attribute is used by most of the other attributes as a time baseline.
- TimeStatus – bitmap that indicates whether or not other Time attributes are read-only and if they synchronized with the time standard.
- TimeZone – the number of seconds offset from the value in attribute 0x0000 that determines the appropriate standard time for a particular time zone.

- DstStart –number of seconds to add to Time to identify the start date of daylight saving time.
- DstEnd –number of seconds to add to Time to identify the end date of daylight saving time.
- DstShift – during the daylight saving time period, this is the number of seconds added to StandardTime to determine LocalTime.
- StandardTime – number of seconds added to Time to adjust for the specified time zone.
- LocalTime – number of seconds added to Time to adjust for the specified time zone and DST.

6.4.2 Client-Side

The client-side of the Time cluster allows a read request for the UTC time and the local time. Click on the button labeled "Update Time Attributes" to see these values. Figure 21 shows a screenshot of this functionality.

Figure 21: Time Cluster, Client-Side



Firmware v1.2.0 COM5

Network: Up EUI64: 00244600000f0070 Device Config: Power Outlets

Device	Node ID	EUI64	Type
Mains Power Outlet	0x1b14	0x00244600000f0070	Router
Combined Interface	0x0000	0x00244600000f006f	Coordinator

- Server Clusters
  - Basic (0x0000)
  - Identify (0x0003)
  - Groups (0x0004)
  - Scenes (0x0005)
  - On Off (0x0006)
- Client Clusters
  - Time (0x000a)
  - OTA Upgrade (0x0019)

Value:

Identify
OFF

On Off
OFF

Time
Update Time Attributes

UTC Time
Thu Feb 20 23:11:08 UTC 2014

Local Time
Thu Feb 20, 2014 15:11:08

Network Up | Channel: 14 | PAN ID: 0xA6D2 | Extended PAN ID: 0xF7E1497D01052E5C | Permit Join Time: 0s

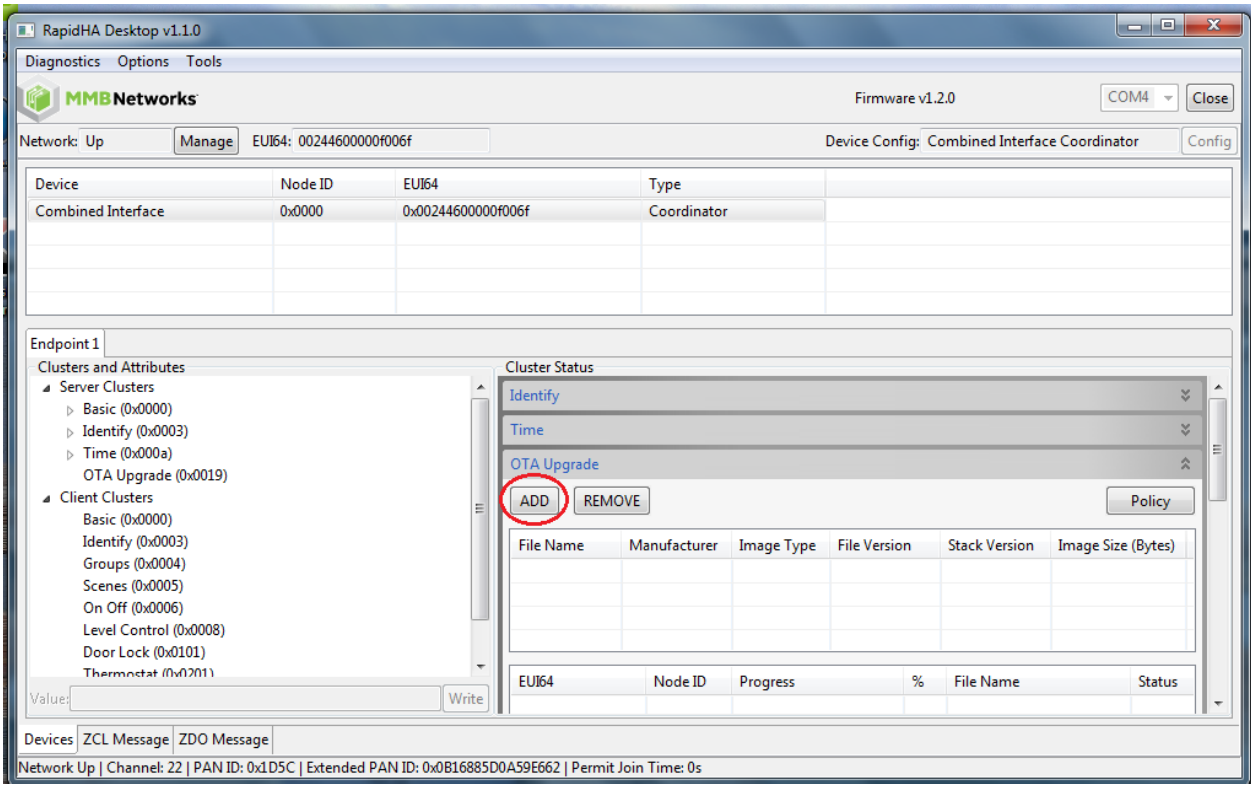
6.5 OTA Upgrade Cluster

RapidHA Desktop allows the coordinator of a network to store a firmware file and then send it wirelessly to an end device on its network. The direction of the OTA firmware upgrade always goes from server to client, more specifically, from coordinator to end device.

6.5.1 Server-Side

To store the firmware file on the coordinator, click the button labeled "ADD" (shown in Figure 22). This action brings up a Windows Explorer window that allows you to browse for and open the appropriate file. A firmware file for OTA transfer will be named "RapidHA_v1.2.0_rc1_prodKey.ota" or something similar.

Figure 22: OTA Upgrade Interface



Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Device	Node ID	EUI64	Type
Combined Interface	0x0000	0x00244600000f006f	Coordinator

File Name	Manufacturer	Image Type	File Version	Stack Version	Image Size (Bytes)

EUI64	Node ID	Progress	%	File Name	Status

Network Up | Channel: 22 | PAN ID: 0x1D5C | Extended PAN ID: 0x0B16885D0A59E662 | Permit Join Time: 0s

After the .ota file is opened it will be identified by name in the Cluster Status area, along with other file information like its size and version number.

The rest of the upgrade process happens automatically, depending on the selections made in the OTA Settings popup window. Click on the button labeled "Policy" to bring up this window. Figure 23 shows both the location of the button and the OTA Settings window.

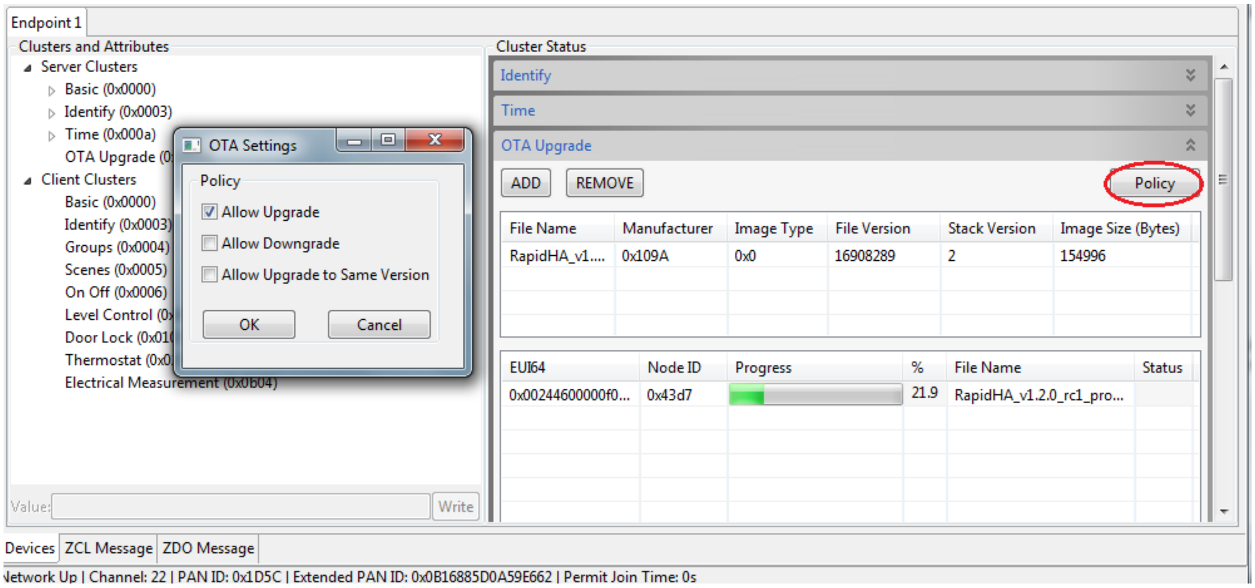
6.5.1.1 Manage OTA Upgrade

There are three selections for managing the automatic upgrade. Choose any number of them, including none of them, which would effectively disallow any OTA upgrade.

Zero or any number of the following options may be checked:

- **Allow Upgrade** – when this option is checked, all end devices on the network will have their firmware upgraded if the version number of the .ota file is higher than the version number of the firmware already on the device.
- **Allow Downgrade** – when this option is checked, all end devices on the network will have their firmware downgraded if the version of the .ota file is lower than the version number of the firmware already on the device.
- **Allow Upgrade to Same Version** – when this option is checked, all end devices on the network will have their firmware updated if the version of the .ota file is the same as the version number of the firmware already on the device.

Figure 23: OTA Upgrade in Progress



OTA Settings

Policy

☑ Allow Upgrade

□ Allow Downgrade

□ Allow Upgrade to Same Version

OK Cancel

File Name	Manufacturer	Image Type	File Version	Stack Version	Image Size (Bytes)
RapidHA_v1....	0x109A	0x0	16908289	2	154996

EUI64	Node ID	Progress	%	File Name	Status
0x00244600000f0...	0x43d7	🟩	21.9	RapidHA_v1.2.0_rc1_pro...	

Network Up | Channel: 22 | PAN ID: 0x1D5C | Extended PAN ID: 0x0B16885D0A59E662 | Permit Join Time: 0s

6.5.1.2 End Device Behavior

The server (coordinator) hosts the file and the client (one of the various end devices) will query the server one minute after joining the network and once every five minutes thereafter. The firmware file is detected automatically and the upgrade will proceed based on the options checked in the OTA Settings popup window. Each upgrade is tracked visually as shown in Figure 23. EUI64 numbers and Node IDs identify the particular end devices being acted upon.

When the firmware upgrade has completed, the Status field will hold one of the values in Table 1.

Table 1: Status of OTA Upgrade

Status	Enumeration	Notes
Success	0x00	The firmware file was successfully transferred to the end device.
Abort	0x95	Server or client decided to abort the upgrade process.
Invalid Image	0x96	Firmware image rejected, e.g., bad CRC.
Require More Image	0x99	Client requires more firmware image files to complete the upgrade successfully.

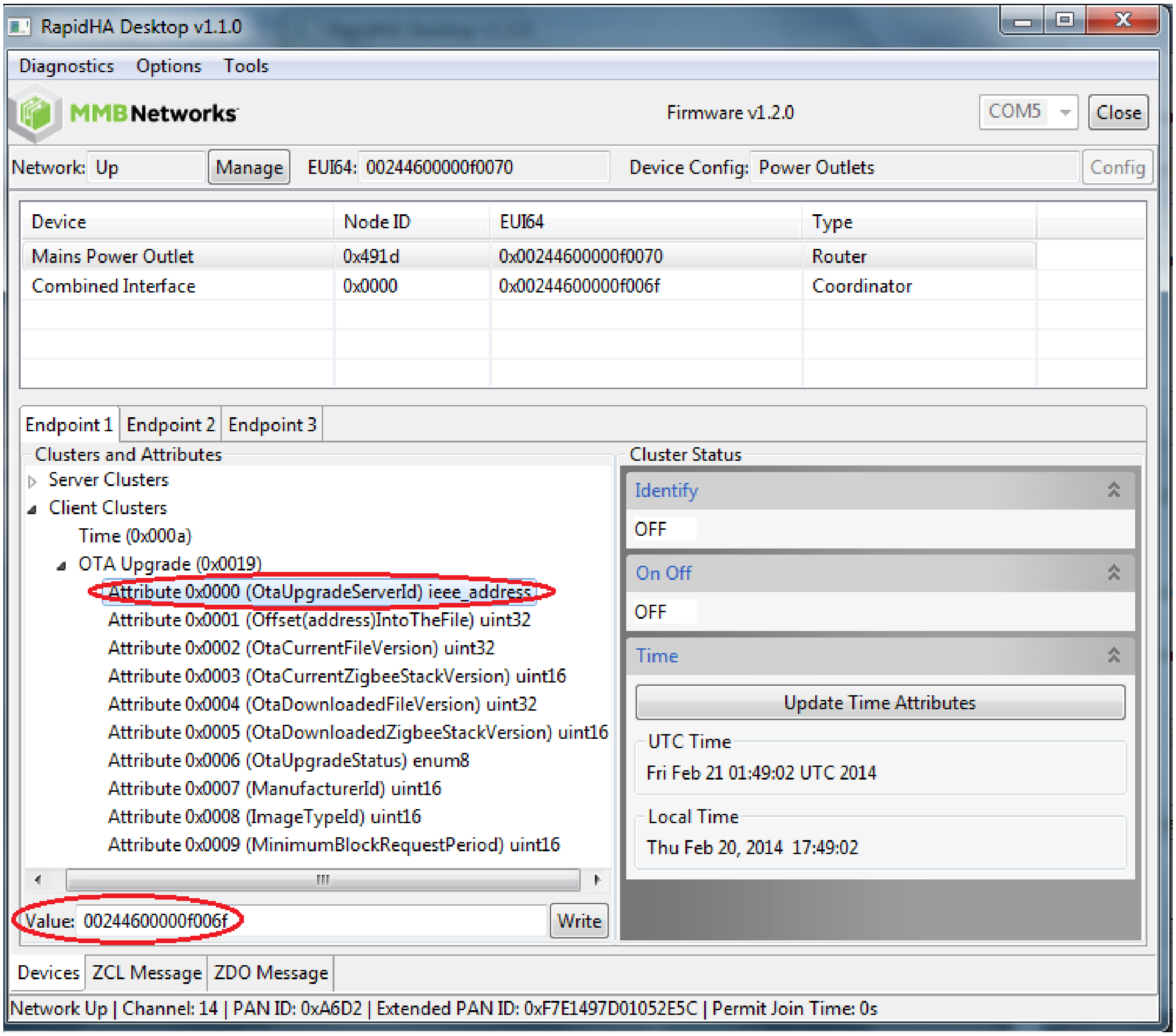
6.5.2 Client-Side

The client-side of the OTA Upgrade cluster allows read/write access to file information about both the current firmware image running on the end device and the upgrade image. This information includes stack and file version numbers, as well as status and progress indicators regarding the over the air transfer of the upgrade image. The status attribute (0x0006) has a non-zero value when the firmware upgrade is in progress.

See Figure 24 for an example of reading the IEEE address of the OTA upgrade server. Click on any attribute to read and display its value.

Please note: do **not** write to client-side attributes while a firmware upgrade is in progress.

Figure 24: OTA Cluster, Client-Side Attributes



Firmware v1.2.0

COM5

Network: Up

EUI64: 00244600000f0070

Device Config: Power Outlets

Device	Node ID	EUI64	Type
Mains Power Outlet	0x491d	0x00244600000f0070	Router
Combined Interface	0x0000	0x00244600000f006f	Coordinator

Clusters and Attributes

- Server Clusters
- Client Clusters
  - Time (0x000a)
  - OTA Upgrade (0x0019)
    - Attribute 0x0000 (OtaUpgradeServerId) ieee_address
    - Attribute 0x0001 (Offset(address)IntoTheFile) uint32
    - Attribute 0x0002 (OtaCurrentFileVersion) uint32
    - Attribute 0x0003 (OtaCurrentZigbeeStackVersion) uint16
    - Attribute 0x0004 (OtaDownloadedFileVersion) uint32
    - Attribute 0x0005 (OtaDownloadedZigbeeStackVersion) uint16
    - Attribute 0x0006 (OtaUpgradeStatus) enum8
    - Attribute 0x0007 (ManufacturerId) uint16
    - Attribute 0x0008 (ImageTypeId) uint16
    - Attribute 0x0009 (MinimumBlockRequestPeriod) uint16

Value: 00244600000f006f

Cluster Status

Identify: OFF

On Off: OFF

Time

Update Time Attributes

UTC Time: Fri Feb 21 01:49:02 UTC 2014

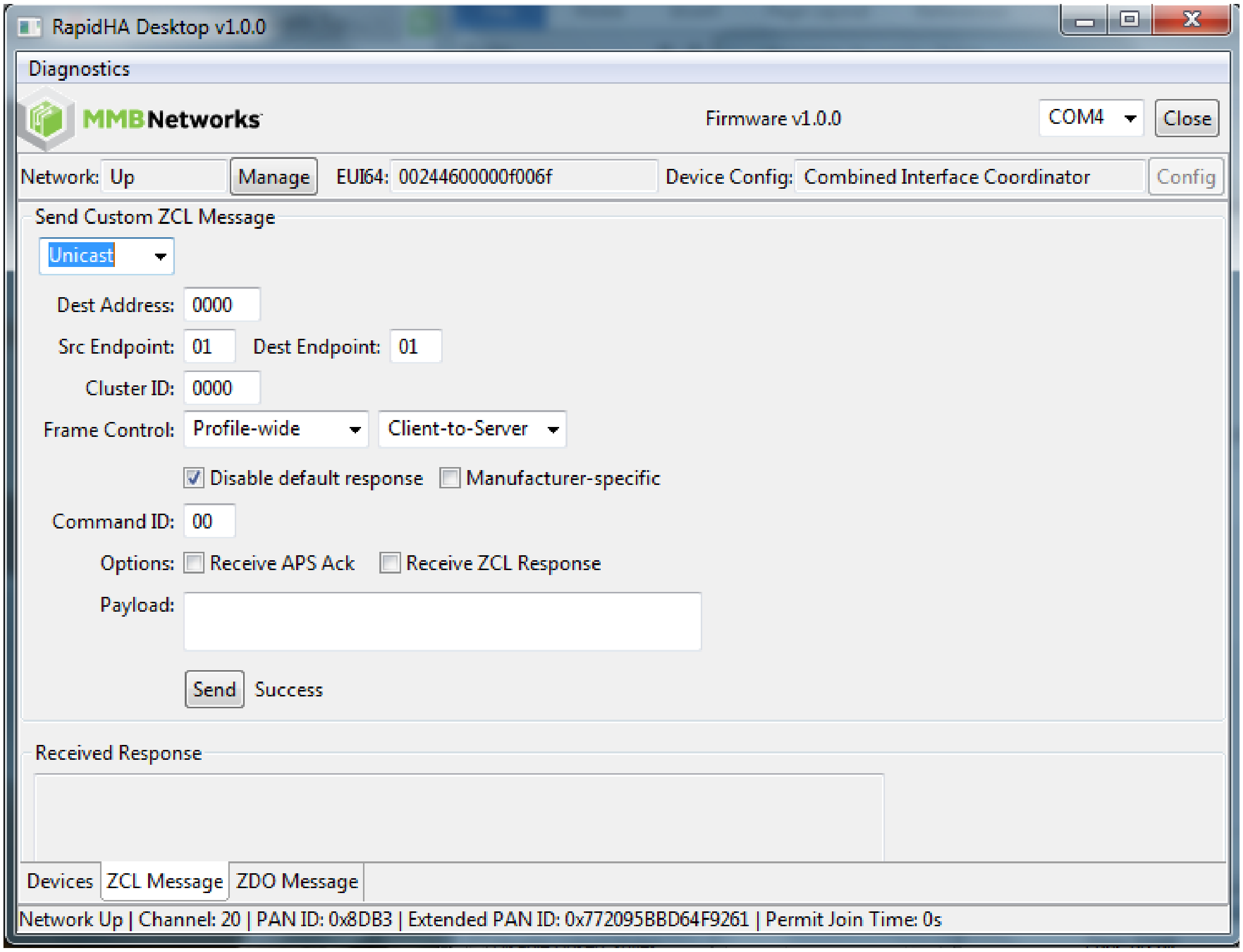
Local Time: Thu Feb 20, 2014 17:49:02

Network Up | Channel: 14 | PAN ID: 0xA6D2 | Extended PAN ID: 0xF7E1497D01052E5C | Permit Join Time: 0s

7.0 ZCL Message Tab

The ZCL Message tab allows you to send any ZigBee Cluster Library (ZCL) command frame with a custom-built payload. See Figure 25 for a screenshot of this tab's default contents.

Figure 25: ZCL Message Tab



Firmware v1.0.0

COM4

Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Send Custom ZCL Message

Unicast

Dest Address: 0000

Src Endpoint: 01 Dest Endpoint: 01

Cluster ID: 0000

Frame Control: Profile-wide Client-to-Server

☑ Disable default response ☐ Manufacturer-specific

Command ID: 00

Options: ☐ Receive APS Ack ☐ Receive ZCL Response

Payload:

Success

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

7.1 Destination Address

Select unicast, multicast, or broadcast from the dropdown menu.

- Unicast – destination address is a Node ID.
- Multicast - destination address is a group ID.
- Broadcast - valid destination addresses are:

  0xFFFC for all routers and coordinators

  0xFFFD for all non-sleepy devices

  0xFFFF for all devices including sleepy

Endpoints are entered as decimal values and Cluster IDs as hexadecimal.

7.2 Frame Control

There are four parameters available for frame control.

1. Frame type - in the dropdown menu select “Cluster-specific” or “Profile-wide”. An example of a cluster-specific command is sending the On command to the On/Off cluster. An example of a profile-wide command is Read Attribute on any cluster that has an attribute.
2. Direction – in the dropdown menu select “Client-to-Server” or “Server-to-Client”. Typically, the server stores attributes and the client makes requests for and manipulates those attributes.
3. Disable default response – when checked this option will only send the default response when there is an error; otherwise it will not be sent.
4. Manufacturer Specific – when checked this option indicates that a command refers to a manufacturer-specific extension to a profile and that there will be a manufacturer code field in the ZCL frame.

7.3 Command ID

The command identifier is an 8-bit value. Its operation depends on the frame type. For example, a value of “00” sent to the On/Off cluster as a profile-wide frame would be processed as a Read Attribute command. If it was sent as a cluster-specific command, the same value of “00” in the Command ID field would be an Off command to the On/Off cluster.

Both profile-wide commands and cluster-specific commands are detailed in the ZigBee Cluster Library specification from the ZigBee Alliance.

7.4 Options

Check one or more of the following options to enable its functionality.

- Receive APS Ack – check this option to receive an acknowledgement from the destination node that the command frame was processed correctly. The Ack will appear in the Response window.
- Receive ZCL Response – check this option to view the response from the destination node. If there is no response defined for the command, a ZCL timeout message will be sent.

  RapidHA Desktop may receive multiple responses to a multicast or broadcast when this option is checked. All responses will be viewable in the Response window.

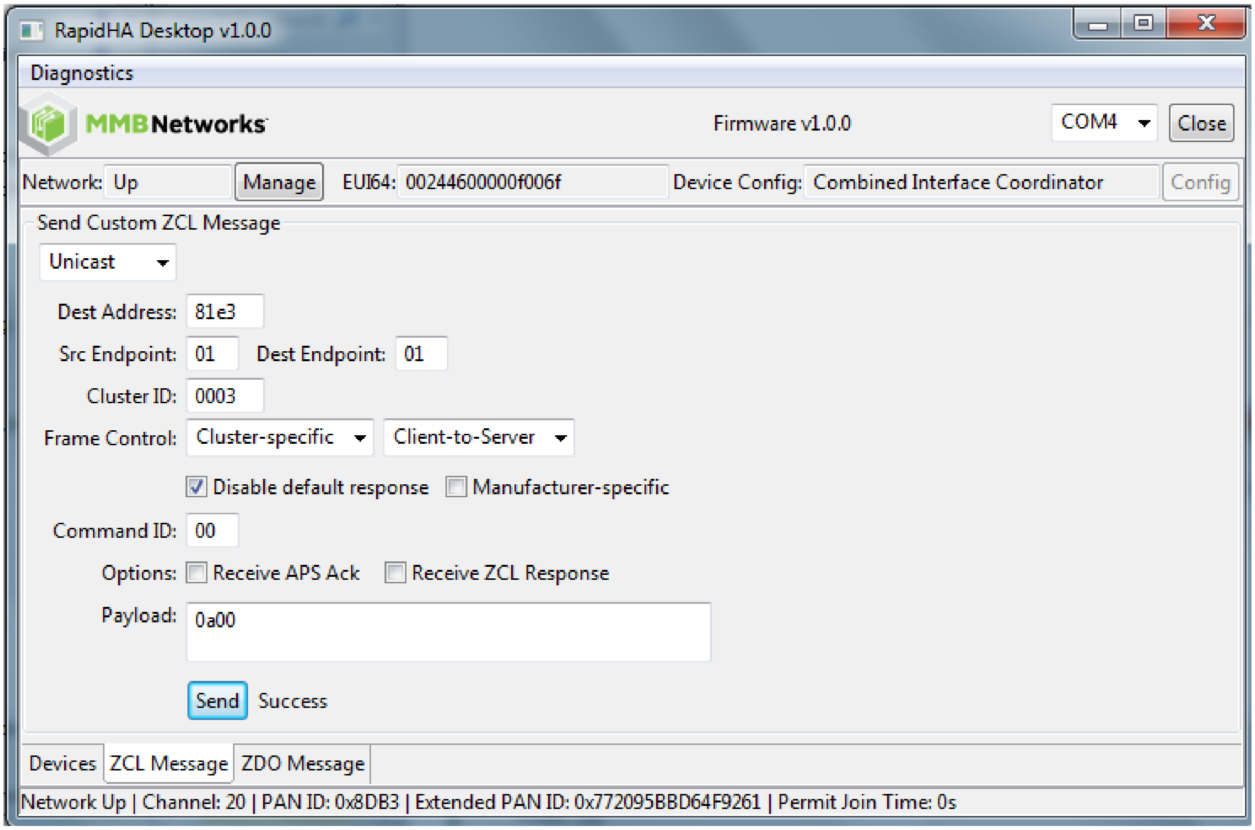
7.5 Payload

Enter field values as hexadecimal bytes ordered by least significant bytes first. The maximum size payload is 108 bytes.

7.6 Example ZCL Message

It is straightforward to send a ZCL message. In Figure 26 is an example of sending the Identify command from the coordinator to node 0x81e3. The command is sent to the Identify cluster (0x0003) with “0a00” in the payload to request the device identify for ten seconds.

Figure 26: Send Identify Command



RapidHA Desktop v1.0.0

Diagnostics

MMBNetworks Firmware v1.0.0 COM4 Close

Network: Up Manage EUI64: 00244600000f006f Device Config: Combined Interface Coordinator Config

Send Custom ZCL Message

Unicast

Dest Address: 81e3

Src Endpoint: 01 Dest Endpoint: 01

Cluster ID: 0003

Frame Control: Cluster-specific Client-to-Server

☑ Disable default response ☐ Manufacturer-specific

Command ID: 00

Options: ☐ Receive APS Ack ☐ Receive ZCL Response

Payload: 0a00

Send Success

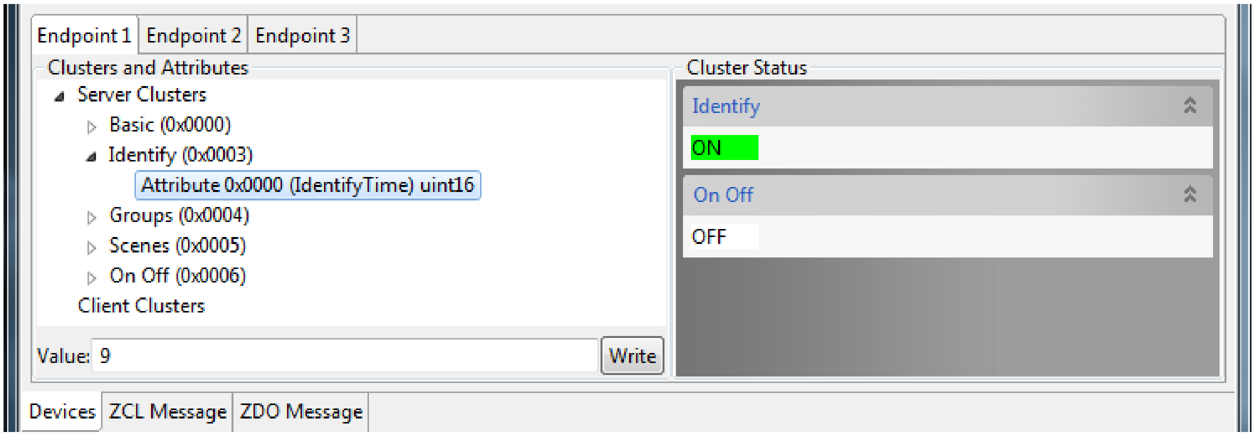
Devices ZCL Message ZDO Message

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

When a device receives an Identify command, the IdentifyTime attribute is updated to reflect the change and the status of the Identify cluster is updated as shown in Figure 27. The status will remain “ON” while the specified time counts down to zero. The attribute can be read by clicking on it and its value will appear in the textbox labeled “Value”. This number reports how much time is left. In Figure 27, it shows that the attribute was read 1 second after the cluster began identification; therefore it shows the time left as 9 seconds.

From the ZCL Message tab, a node can send any ZCL message to any other node in the network, including sending a message to itself.

Figure 27: Identify Status Updated



Value: 9

Identify: ON

On Off: OFF

8.0 ZDO Message Tab

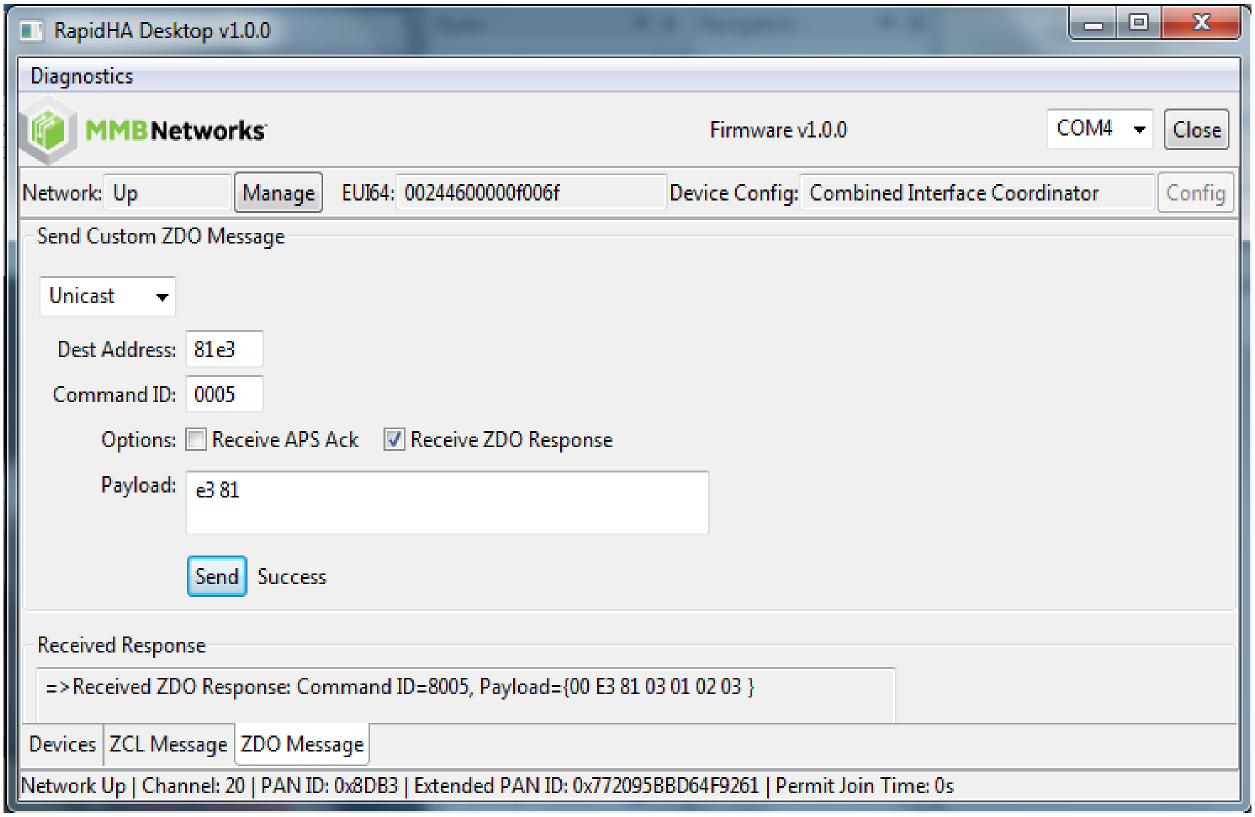
The ZDO Message interface lets you send any ZDO unicast or broadcast message. The ZDO (ZigBee Device Object) runs on endpoint 0 on every ZigBee compliant device.

To send a ZDO message, select “Unicast” or “Broadcast” from the dropdown menu and then set the following parameters:

- Dest Address – the destination address is the network address (a.k.a., Node ID) assigned when the device joined the network. The only valid broadcast destination is 0xFFFD (all non-sleepy devices).
- Command ID – these values are defined in the ZigBee Specification and are referred to as ClusterID numbers.
- Payload – the fields in the payload are hex bytes ordered by least significant bytes first. The maximum payload size is 114 bytes.

The example screenshot in Figure 28 shows the information used to send a ZDO Active Endpoint Request (Command ID 0005) to node 0x81e3.

Figure 28: Unicast ZDO Command



Network: Up

EUI64: 00244600000f006f

Device Config: Combined Interface Coordinator

Unicast

Dest Address: 81e3

Command ID: 0005

Options: □ Receive APS Ack ☑ Receive ZDO Response

Payload: e3 81

Success

=>Received ZDO Response: Command ID=8005, Payload={00 E3 81 03 01 02 03 }

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

The "Dest Address" is sent, little endian, as the payload of the command. This is because the format of the command requires it to be included, which would be useful if you wanted to inquire of a parent the active endpoints of one of its children. ZDO command formats are defined in the ZigBee Specification

The Active Endpoint Request command triggers the Active Endpoint Response command, which is reported in the Received Response window if "Receive ZDO Response" is checked. In this example, the payload is {00 E3 81 03 01 02 03}. The interpretation of these byte values is explained in Table 2.

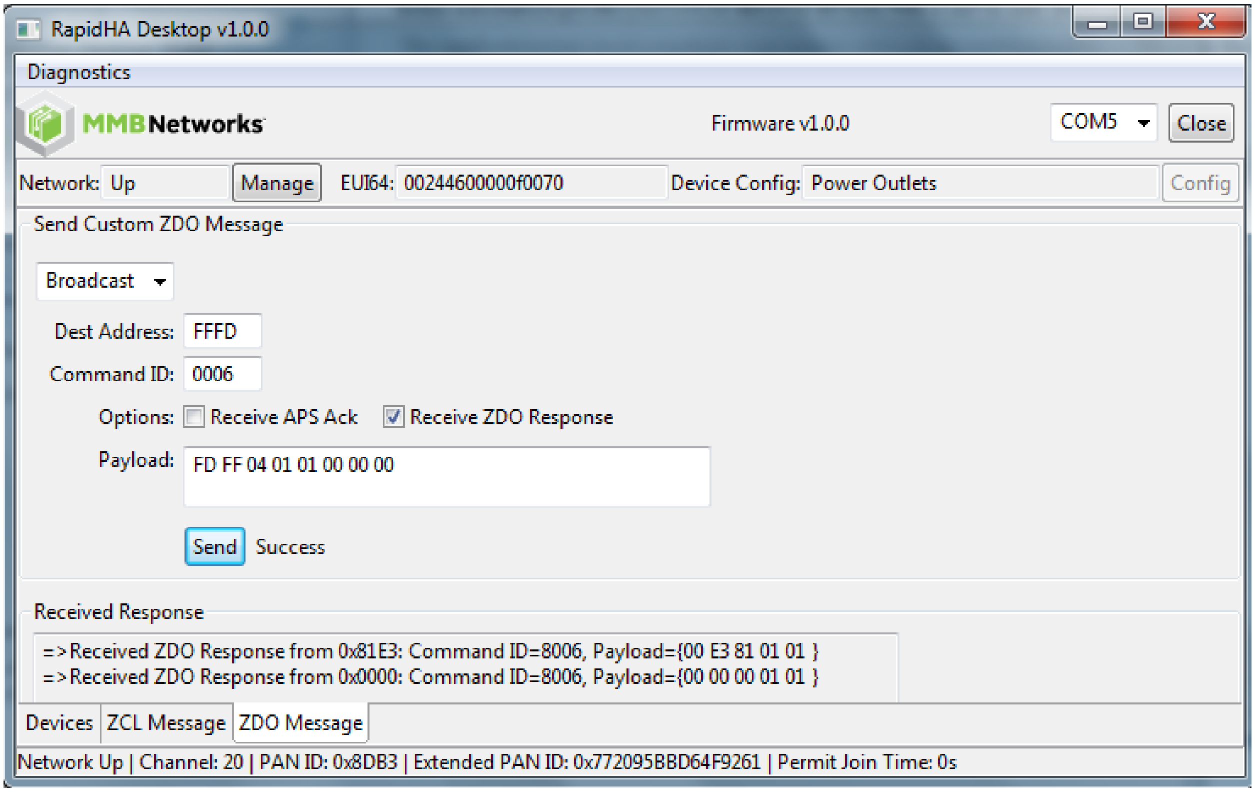
Table 2: Payload Description for Active EP Response Command (0x8005)

Value	Name	Description
00	Status	The command was successful.
E3 81	Node ID	The node identified by "Dest Address" in the Active Endpoint Request command, sent little endian, i.e., least significant byte first.
03	Active EP Count	The number of endpoints defined on the node in question. The screenshot in Figure 27 shows there are, in fact, 3 endpoints on the device with Node ID 0x81e3.
01, 02, 03	Active EP List	The endpoint Id numbers

The screenshot in Figure 29 shows an example of how to broadcast a Match Descriptor Request on the home automation profile (0x0104) to discover devices with a server-side Basic cluster (0x0000).

The application may receive multiple responses to a broadcast and they will all be shown in the Response window when the option “Receive ZDO Response” is checked.

Figure 29: Broadcast to all Non-Sleepy Devices



Firmware v1.0.0

COM5

Network: Up

EUI64: 00244600000f0070

Device Config: Power Outlets

Send Custom ZDO Message

Broadcast

Dest Address: FFFD

Command ID: 0006

Options: □ Receive APS Ack ☑ Receive ZDO Response

Payload: FD FF 04 01 01 00 00 00

Success

Received Response

```
=>Received ZDO Response from 0x81E3: Command ID=8006, Payload={00 E3 81 01 01 }
=>Received ZDO Response from 0x0000: Command ID=8006, Payload={00 00 00 01 01 }
```

Network Up | Channel: 20 | PAN ID: 0x8DB3 | Extended PAN ID: 0x772095BBD64F9261 | Permit Join Time: 0s

9.0 Support

Please email the support team at support@mmbnetworks.com if you have any questions about RapidHA Desktop.

10.0 Revision History

Version	Description	Date	Modified by
0.9.3	Initial Draft	August 21, 2013	Debra Knight
1.0.0	New GUI Two USB sticks operating concurrently Configuration files	December 12, 2013	Debra Knight
1.2.0	Firmware upgrade, serial and OTA; Time cluster	February 24, 2014	Debra Knight