

# EM260 User Guide

**For the EM260 Development Kit and  
JumpStart Edition for the EM260**

**13 November 2008**

**120-4027-000F**



Ember Corporation  
47 Farnsworth Street  
Boston, MA 02210  
+1 (617) 951-0200  
[www.ember.com](http://www.ember.com)



Copyright © 2008 by Ember Corporation

All rights reserved.

The information in this document is subject to change without notice. The statements, configurations, technical data, and recommendations in this document are believed to be accurate and reliable but are presented without express or implied warranty. Users must take full responsibility for their applications of any products specified in this document. The information in this document is the property of Ember Corporation.

Title, ownership, and all rights in copyrights, patents, trademarks, trade secrets and other intellectual property rights in the Ember Proprietary Products and any copy, portion, or modification thereof, shall not transfer to Purchaser or its customers and shall remain in Ember and its licensors.

No source code rights are granted to Purchaser or its customers with respect to all Ember Application Software. Purchaser agrees not to copy, modify, alter, translate, decompile, disassemble, or reverse engineer the Ember Hardware (including without limitation any embedded software) or attempt to disable any security devices or codes incorporated in the Ember Hardware. Purchaser shall not alter, remove, or obscure any printed or displayed legal notices contained on or in the Ember Hardware.

Ember, Ember Enabled, EmberZNet, EmberZNet PRO, InSight, and the Ember logo are trademarks of Ember Corporation.

All other trademarks are the property of their respective holders.



---

## Contents

<b>1</b>	<b>About this Guide</b> .....	<b>5</b>
	Purpose .....	5
	Audience .....	5
	Documentation Conventions.....	6
	Getting Help .....	6
<b>2</b>	<b>Introducing Ember Development Kits</b> .....	<b>7</b>
	Development Kit Contents.....	8
	Hardware Requirements .....	8
	Software Requirements .....	8
	Before You Begin .....	8
<b>3</b>	<b>Development Kit Components</b> .....	<b>9</b>
	Hardware .....	9
	Software .....	13
	Documentation .....	17
<b>4</b>	<b>Setting Up Hardware Components</b> .....	<b>18</b>
	InSight Adapter Settings.....	19
	Hardware Connections .....	20
	EUI-64 Assignments.....	20
	Attaching and Removing Breakout Board Modules.....	20
	Powering the Breakout Board .....	20
	Resetting InSight Adapters .....	21
	Connecting the SMA Connector to the RCM.....	21
	Battery Powering Modules .....	21
	Configuring the Serial Communication Path .....	22
<b>5</b>	<b>Development Kit Components</b> .....	<b>23</b>
	Positioning Network Nodes.....	23
<b>6</b>	<b>Using the InSight Adapter</b> .....	<b>24</b>
	InSight Adapter Default Configuration .....	24
	Connecting to an InSight Adapter Socket.....	24
	Monitoring Radio Communication Module Data.....	25
	Connecting to the USB Administrative Port .....	25

	Resetting an InSight Adapter Socket.....	26
	Resetting the InSight Adapter .....	26
	Terminating a Connection to the Administrative Socket .....	26
	Configuring the InSight Adapter .....	26
	Getting Help on InSight Adapter Commands .....	27
	Uploading Software to the InSight Adapter .....	27
<b>7</b>	<b>InSight Adapter Commands .....</b>	<b>28</b>
	Syntax Conventions .....	28
	Entering Commands .....	28
	Connecting to the InSight Adapter Administrative Port .....	28
	InSight Adapter Commands.....	28

---

# 1

## About this Guide

### Purpose

This document describes the Ember Development Kit and explains how to set up its hardware and use the Ember InSight Adapter.

- For information on developing and debugging applications with InSight Desktop, see its online documentation.
- For information on designing and debugging an application based on Ember's software stack EmberZNet, refer to the *EmberZNet Application Developer's Guide* (120-4028-000), *EmberZNet Application Developer's Reference Manual* (120-3021-000), and the *EM260 Datasheet* (120-0260-000).
- Other documents of interest include:

Document	Part No.
<i>Quick Start Guide</i>	120-4026-000
<i>EM260 Breakout Board Technical Specification: SPI/UART, 128kB Version</i>	120-2008-000
<i>EM260 Radio Communication Module Technical Specification: SPI/UART Version</i>	120-2007-000
<i>InSight Adapter Technical Specification</i>	120-2002-000

### Audience

This document is intended for embedded software engineers who are responsible for developing wireless applications with the Ember Development Kit.

## Documentation Conventions

Notation	Meaning	Example
<i>Italics</i>	Identifies on-screen software menu options.	<i>Connect</i>
UPPERCASE	Identifies a keyboard key.	ENTER
Right-angle bracket	Delimits a series of software program menu options to be clicked.	<i>Open &gt; Save</i>
Courier	Identifies software code and, in body text, variables.	<code>void Main(String[] argv) the buffer variable</code>
<i>Courier Italics</i>	Identifies a variable name.	<i>ipAddress</i>

## Getting Help

Development Kit customers are eligible for training and technical support. You can use the Ember web site [www.ember.com](http://www.ember.com) to obtain information about all Ember products and services, and to sign up for product support. You can also contact Ember technical support at <http://portal.ember.com>.

If you have any questions about your Development Kit, contact your Ember account representative at one of the following locations:

<b>United States</b>	47 Farnsworth Street Boston, MA 02210 Telephone: 617-951-0200 Fax: 617-951-0999
<b>Asia/Pacific</b>	HK Spinners Industrial Bldg, Phase 5 5/F Flat D 760-762 Cheung Sha Wan Rd Kowloon Hong Kong Telephone: +852-8120-5375
<b>Europe</b>	Unit 300 Science Park Milton Road Cambridge CB4 0XL, UK Telephone: 44 (0) 1223 423322 Fax: 44 (0) 1223 423390

---

# 2

## Introducing Ember Development Kits

The Ember Development Kit is used for application design and development. The kit contains a variety of hardware for quickly creating a prototype of the customer's hardware, EmberZNet™, Ember's implementation of the ZigBee Pro stack software specification, and various development tools which enable the development of the customer's application software.

EmberZNet runs on top of IEEE 802.15.4-compliant radio and link layer software. EmberZNet is a self-organizing, self-healing mesh networking protocol stack. Combined with Ember's powerful IEEE 802.15.4 compliant radios, the EmberZNet network stack provides complete networking services from the physical layer up to a reliable application profile support layer. The stack produces networks that are reliable, flexible, secure, and easy to use.

**Note:** If you have not yet read the *EM260 Quick Start Guide* (120-4023-000), please do so before reading further in this User Guide.

EmberZNet<sup>1</sup> provides a common application programming interface (API) that utilizes the underlying layers. This API provides support for the following layers of the OSI model:

- **PHY:** radio control
- **MAC:** medium access
- **LINK:** route discovery
- **NETWORK:** routing, association
- **APPLICATION:** Application Profile Support (APS) for reliable message delivery

The EmberZNet stack can be used to build ZigBee-based applications. Applications that are built on this stack benefit from the following features:

- PHY and MAC layers comply with the IEEE 802.15.4 standard
- LINK and NETWORK layers comply with the latest ZigBee specification
- The APS layer provides the interface to the application for data transfer and management, including reliable transport and binding.

The Ember Development Kit includes InSight Desktop, which facilitates application development and debugging. InSight Adapters incorporate Power-over-Ethernet (PoE) capability, which can simplify node deployment in the test environment. The boards also provide a direct connection for debugging customer-designed hardware. For detailed information about Ember's customer hardware debug interface, see the *EM260 Datasheet* (120-0260-000).

---

<sup>1</sup> The release version of EmberZNet at the time of this publication is 3.3.

## Development Kit Contents

Depending on your Ember development kit, you will have the specified count for each component as summarized in Table 1.

Table 1. User Guide components for EM260 Development Kit and JumpStart Edition for EM260

Component	EM260 Development Kit	JumpStart Edition for EM260
Installation CD	1	1
Breakout boards	8	3
InSight adapters	8	3
Radio communication modules	8	3
Battery packs	8	3
12V DC power supplies	8	3
Power-over-Ethernet switch	1	1
6-foot USB cable	1	1
InSight Data Emulation Interface (DEI) interfaces	8	3
InSight Port cables	8	3
2-foot category 5e cables	8	3
Murata switch connector adapter	1	1

## Hardware Requirements

- Workstation with Ethernet connectivity
- Network hub with DHCP server

## Software Requirements

- Microsoft Windows XP or Microsoft Windows Vista (XP preferred)
- Java Runtime Environment, version 1.5 or higher
- Adobe Acrobat Reader (available free from <http://www.adobe.com/products/acrobat/readstep2.html>)

## Before You Begin

The Ember Development Kit is designed only for the operating conditions and interfaces that are described in the *EM260 Breakout Board Technical Specification: SPI/UART, 128kB Version (120-2008-000)*. Any modifications or alterations to the hardware are liable to cause irreparable damage to the Development Kit, and shall render its warranty null and void.

If you wish to use the Ember Development Kit in a non-standard fashion, consult with Ember Support or Ember Sales before starting development. This is especially true if you are embarking on an aggressive development cycle, plan to use custom hardware, or wish to use alternate software configuration files, images, hardware configurations, or bit files. Contact Ember and describe your plan, so we can help you minimize the risk of delays in your development cycle.

---

# 3

## Development Kit Components

This chapter describes EM250 Development Kit components in three sections:

- Hardware
- Software
- Documentation

### Hardware

The EM260 Development Kit has eight (8) pieces each of three major hardware components:

- Breakout board
- Radio communication module
- InSight Adapter

The JumpStart Edition for the EM260 development board set has three (3) pieces each of the same major components.

In addition, the EM260 Development Kit has the following hardware components:

- AC power adapter
- Power-Over-Ethernet (PoE) switch

### Breakout board

Breakout boards (see Figure 1) serve as test beds for network hardware and applications. Each breakout board has an associated radio communication module (RCM), which can be deployed as a node within a network application. Each breakout board connects with its InSight Adapter (ISA) through two communication links: the InSight Port and the ISA's debug port. This allows the breakout board's RCM to interface with InSight Desktop (ISD) through the ISA's InSight Port and its wired Ethernet connection. These interconnections are described later in this chapter.

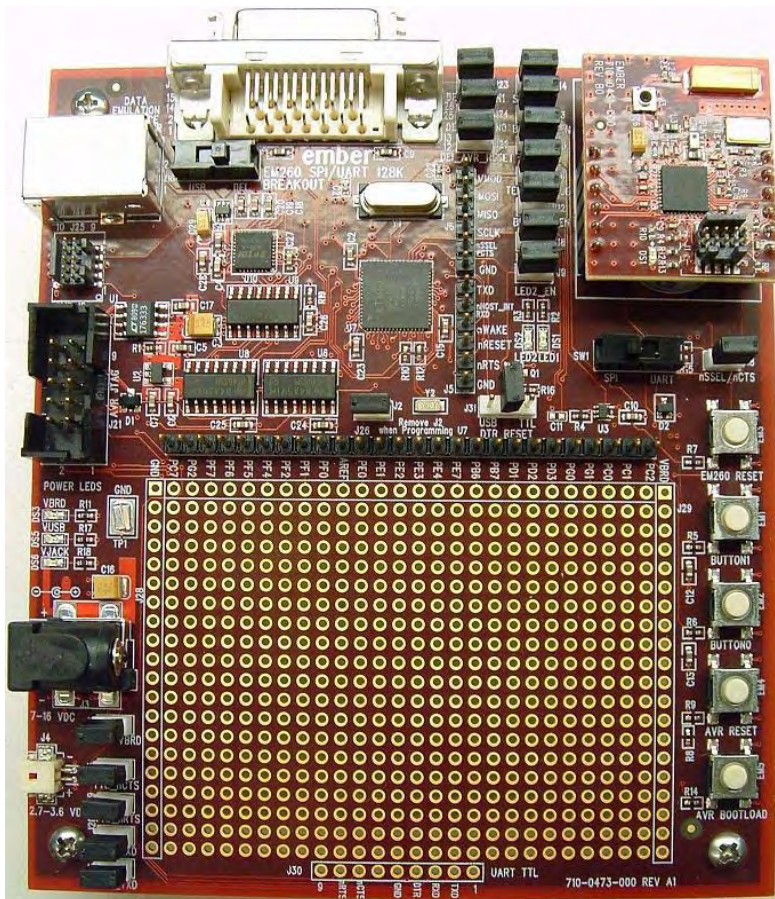


Figure 1. Breakout Board

The breakout board (Figure 1) has the following components:

- Host microcontroller, ATmega128L from Atmel
- AVR ISP- and JTAG-shrouded connectors for programming and debug
- Configurable hardware support for application development
  - Temperature sensor
  - Two buttons
  - Piezo buzzer
  - Two LEDs
- Selectable USB connector with RS-232 transceiver
- EM260 SPI/UART RCM reset button
- Host microcontroller reset button
- Host microcontroller bootloader button
- 2" x 2.6", 0.1" pitch prototyping area
- 26-pin, 0.1" pitch, single-row header for access to all host GPIO signals
- 2 6-pin, 0.1" pitch, single-row EM260 SPI/UART RCM connectors
- 2 6-pin, 0.1" pitch, single-row headers for access to EZSP signals
- 9-pin, 0.1" pitch, single-row header footprint for access to 3.3V TTL-compliant UART signals
- 26-pin, low-profile InSight data emulation interface with configuration header

- Automatic DC power source selection (12V DC wall wart, 5V USB bus power, or AAA battery pack)
- Power indicator LEDs for 12V DC wall wart, 5V USB bus power, and board power
- Jumper separation of module and board power for accurate module current measurements

**Warning:** To use the InSight Data Emulation Interface, you must power the radio communication module from an InSight Adapter that is powered by its Ethernet connection or from the Development Kit's 12V DC power supply. Powering the radio communication module from the battery pack or USB input when the InSight Data Emulation Interface is in use can damage the InSight Adapter.

For detailed information about the breakout board, see the *EM260 Breakout Board Technical Specification: SPI/UART, 128kB Version (120-2008-000)*.

## Radio communication module

The Development Kit Radio Communication Module (RCM) shown in Figure 2 is used together with the Development Kit breakout board to develop and debug application software. Its low-power design lets you remove it from the breakout board when the application development phase ends. You can then power the module with a battery and place it into the desired product enclosure for deployment.

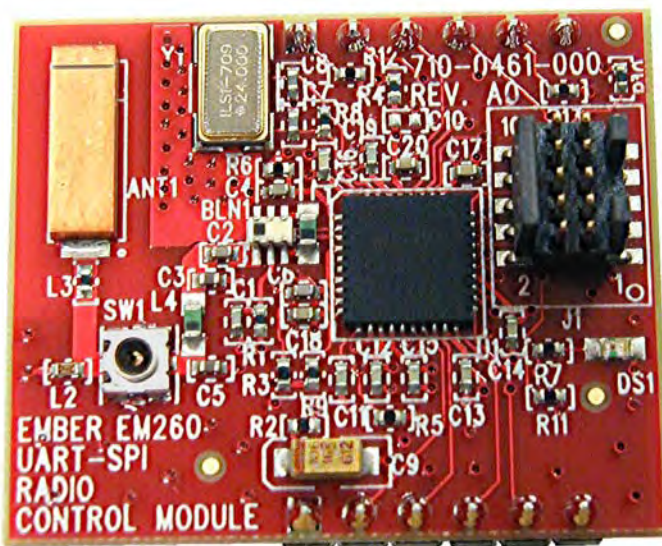


Figure 2. EM260 RCM

The radio communication module includes these components:

- EM260 system:
  - EM260 IEEE 802.15.4-compliant, 2.4 GHz radio transceiver and network processor
- Murata RF switch connection
- Two LEDs
- InSight Port connector

Each radio communication module is assigned a unique IEEE 64-bit identifier (EUI-64). This number is printed on a label affixed to the bottom of the module and can also be viewed in InSight Desktop. For detailed information about the radio communication module supplied with your kit, see the *EM260 Radio Communication Module Technical Specification: SPI/UART Version (120-2007-000)*.

## InSight Adapter

InSight Adapter connects the breakout board to the Ethernet. Each adapter transmits network data collected by InSight Port and conveys it over its Ethernet connection to InSight Desktop. It also picks up any messages or new software that are addressed to this breakout board, and processes emulation and debug commands.

The InSight Adapter has the following components:

- InSight Port interface to the radio communication module, providing programming and debugging services.
- TCP/IP 10/100 Ethernet interface with Power-over-Ethernet functionality.

## AC power adapter

**Caution:** Use only the AC power adapter provided in your Development Kit. Any other power adapter is liable to severely damage the radio communication module microprocessor.

The AC power adapter (Figure 3) is a linear regulated wall plug power supply with 12V DC at 1600mA regulated output. It has a six-foot output cord and a power supply. Four interchangeable snap-in input plugs are provided with various port configurations suitable for various locales: North America, Europe, United Kingdom, and Australia. The Development Kit cable complies with local regulations.



Figure 3. AC power adapter with interchangeable AC plugs

Table 2 lists all of the components and the acceptable combinations of power sources that yield normal operation. Potential power sources include the following:

- Power-over-Ethernet (PoE)
- External AC Adapter (similar to the example in Figure 3)
- Battery pack (3VDC)
- Power over InSight Port on ISA (the RCM Power Select switch on the front panel must be set to *Int.*)
- USB cable
- From RCM dedicated power

Table 2. Safe combination of operating power

	PoE	Ex AC Adapter	Battery	ISA Switch	USB	Breakout Board
ISA	X	X	N/A	N/A	N/A	N/A
EM260	N/A	N/A	X <sup>2</sup>	X	N/A	X
EM260 Breakout Board	N/A	N/A	X	X <sup>3</sup>	X	N/A

## Power-Over-Ethernet (PoE) switch

The EM260 Development Kit contains a Netgear 16-port Power-over-Ethernet (PoE) switch that supplies power to an InSight Adapter over a standard TIA/EIA-568 category 5 cable. (The JumpStart Edition for the EM260 contains a Netgear 8-port (PoE) switch.) The InSight Adapter can, in turn, supply power to an Ember development board and the radio communication module that is mounted on it. Thus, you can place breakout boards wherever an Ethernet cable connection is available.

The DC operating power for data terminal units is fed through unused pairs 7/8 and 4/5. For more information about the Netgear switch, see the online technical specifications.

The Netgear switch that is supplied in the EM260 Development Kit is certified by Ember. If you wish to use a different PoE switch, please consult with Ember technical support: [support@ember.com](mailto:support@ember.com)

## Software

The Ember Development Kit contains the following software components:

- Network stack
- Hardware Abstraction Layer for the ATmega128LL
- EmberZNet Serial Protocol (EZSP)
- Sample applications and pre-built images
- InSight Desktop

## Network stack

The EmberZNet network stack is an advanced implementation of the Zigbee Pro wireless protocol stack which provides mesh routing algorithms for creating reliable, flexible, and secure networks. The stack's APS layer provides interfaces for delivering messages reliably. The network stack API is documented in the *EmberZNet Application Developer's Reference Manual* (120-3021-000), the *EZSP Reference Guide* (120-3009-000), and the API References for the EM260, available in HTML and PDF versions.

## Hardware Abstraction Layer for the ATmega128L

The hardware abstraction layer (HAL) for the ATmega128L acts as a conduit between the EZSP interface and ATmega microcontroller. Separating the application program from its hardware

---

<sup>2</sup> There is no dedicated power connector available. Power must be hard-wired to the RCM interface connector.

<sup>3</sup> Power is sourced from the RCM module via the InSight Port connected to the ISA with its RCM Power Switch set to *Int*.

implementation enables easy portability. HAL source code is provided so you can customize, extend, or reduce device functionality across different hardware platforms.

The HAL API is documented in the *EmberZNet Application Developer's Guide* and the online HAL API References.

## EmberZNet Serial Protocol (EZSP)

The EmberZNet Serial Protocol (EZSP) is designed to facility communication between a host microcontroller and the EM260 across a serial interface. EZSP is modeled on the EmberZNet 3.x stack, and most API functions are identical to those in EmberZNet 3.x. Differences between the two generally due to timing differences that arise from running the application on a separate processor across a serial interface. For detailed information about EZSP, see the *EZSP Reference Guide* (120-3009-000).

## Sample applications and pre-built images

The EM260 Development Kit contains several source code application examples and a number of pre-built images, which are described in Table 3, Table 4, and Table 5. Images for the host processor are provided in `.bin` files (for ATmega32L host), `.ebl` files (for ATmega128L host) and `.hex` files; images for the EM260 are provided in `.hex` files and `.ebl` files. All applications examples are in the distributions `app` directory and the pre-built images are in the distribution's `build` directory. Expanded descriptions of included sample application code can be found in the `sampleApps.html` page included in the "app" subdirectory of the EmberZNet software installation for EM260.

Table 3. Source Code Application Examples

Application name	Directory Name	Description
Standalone Bootloader	<code>\app\standalone-bootloader-demo-host</code>	Example application that shows how to use the standalone bootloader via SPI, UART or radio.
Manufacturing Library	<code>\app\mfglib-host</code>	Example of how to use the manufacturing library co-resident with your application.
Home Automation	<code>\app\ha</code>	The Home Automation example shows how to use the ZigBee compliant HA- related clusters and build a basic HA application framework. Also includes support for Zigbee Smart Energy (SE) devices and clusters.
Sensor/sink	<code>\app\sensor-host</code>	Aggregation example that shows how a single device collects data from multiple device. In this simple application, sensor nodes collect data and send it periodically to a sink node. The default application preloaded on kit hardware.
ZigBee Device Object	<code>\app\zdo</code>	Example of how to use the Ember Zigbee Device Object library.

Table 4. Pre-built images for ATmega128L Processor

Application name	Filename	Description
HAL test	Atmega128-haltest.{hex,bin}	Ember Hardware Abstraction Layer test program for the ATmega128L.
ATmega128L Bootloader	Bootloader-avr128.hex	Bootloader for ATmega128L
Mobile Sensor	Mobile-sensor.{hex,bin}	Mobile sensor component of the sensor/sink application suite.
Sensor host	Sensor-host.{hex,bin}	Host sensor component of the sensor/sink application suite.
Sink host	Sink-host.{hex,bin}	Sink host component of the sensor/sink application suite.
Sleepy Sensor	Sleepy-sensor-host.{hex,bin}	Sleepy sensor component of the sensor/sink application suite.
Training host	Training-host-avr.{hex;bin}	EZSP Training application for experimenting with EZSP commands in an ASCII-based command line interpreter. See "training-app.html" page included in "app" subdirectory for details.

Table 5. Pre-built images for EM260 network co-processor

Application name	Filename	Description
EM260 Stack images	em260-spi-with-standalone-bootloader.{hex, ebl}	EmberZNet PRO stack for use with EZSP SPI. HEX file contains EM260 Standalone Bootloader with SPI- and radio-based update capability. EBL file for use with standalone-bootloader-spi-em260.hex.
	em260-uart-xon-off-with-serial-bootloader.{hex, ebl}	EmberZNet PRO stack for use with EZSP UART with software (XON/XOFF) flow control. HEX file contains EM260 Standalone Bootloader with UART-based update capability. EBL file for use with standalone-bootloader-uart-em260.hex.
	em260-uart-rts-cts-with-serial-bootloader.{hex, ebl}	EmberZNet PRO stack for use with EZSP UART with hardware (RTS/CTS) flow control. HEX file contains EM260 Standalone Bootloader with UART-based update capability. EBL file for use with standalone-bootloader-uart-em260.hex.
Rangetest	em260-rangetest-with-standalone-bootloader.{hex, ebl}	Rangetest application for the EM260. HEX file contains EM260 Standalone Bootloader with SPI- and radio-based update capability. EBL file for use with standalone-bootloader-spi-em260.hex.
Bootloaders	Standalone-bootloader-spi-em260.hex	SPI version of standalone bootloader, including radio-based update capability.
	Standalone-bootloader-uart-em260.hex	UART version of standalone bootloader.

## InSight Desktop

InSight Desktop (ISD) is a graphical tool that displays network and node activity in real time (see Figure 4). It provides a rich and flexible interface to Ember embedded networks, which helps you develop and debug new network applications.

InSight Desktop includes these features:

- Multiple editor panes that provide tiered views of network activity, letting you drill down to a high-level map of node interactions to the details of each packet.
- Customizable filters that let you specify exactly which network activities to display.
- Log files that save captured data, so you can step through transactions and events for detailed analysis.
- A file browser that lets you easily upload new applications to any connected node.
- A browser-based interface for automatic discovery of Ethernet-connected adapters, and easy management of adapter applications.

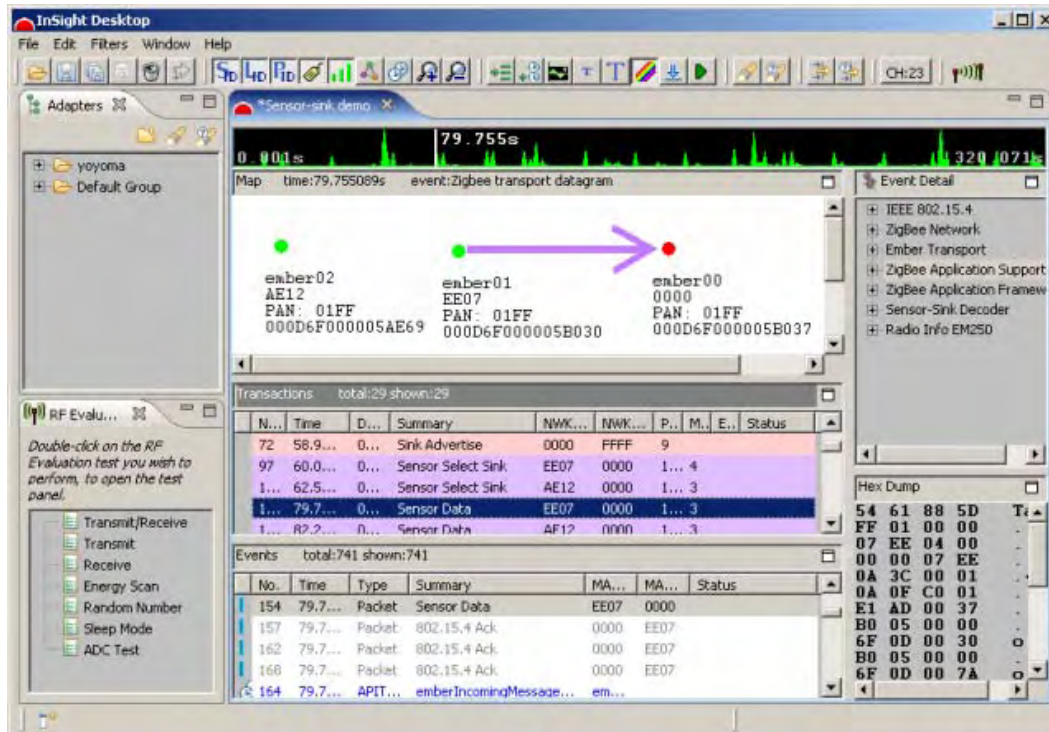


Figure 4. InSight Desktop

For detailed information about InSight Desktop, refer to the online *InSight Desktop User's Guide* and ISD's online help.

## Documentation

References to many Ember documents occur throughout this guide. These documents, and others, are located on the software CD supplied with your Development Kit. Many of these documents are provided in Adobe Acrobat (.PDF) format. You can view these documents online or you can print them out as needed. Any version of the Acrobat Reader Version 8 or later can be used for this purpose. A copy of the free Acrobat Reader can be downloaded at [www.adobe.com/products/acrobat/readstep2.html](http://www.adobe.com/products/acrobat/readstep2.html).

Documentation supplied with this kit includes the following documents and more:

- *EM260 Datasheet* (120-0260-000)
- *EmberZNet Application Developer's Guide* (120-4028-000)
- *EmberZNet Application Developer's Reference Manual* (120-3021-000)
- *InSight Adapter Technical Specification* (120-2002-000)
- *EM260 Breakout Board Technical Specification: SPI/UART, 128kB Version* (120-2008-000)
- *EM260 Radio Communication Module Technical Specification: SPI/UART Version* (120-2007-000)
- *EmberZNet API Reference: For the EM260 Co-Processor* (120-3020-000)—PDF version
- Application Notes
- ISD Online Help
- Pre-built images and hardware designs
- Product licensing information

---

# 4

## Setting Up Hardware Components

This chapter contains the following sections:

- InSight Adapter Settings
- Hardware Connections
- EUI-64 Assignments
- Attaching and Removing Breakout Board Modules
- Powering the Breakout Board
- Resetting InSight Adapters
- Connecting the SMA Connector to the RCM
- Battery Powering Modules
- Configuring the Serial Communication Path

The schematic in Figure 5 provides a high-level view of Development Kit components and connections, using power over Ethernet.

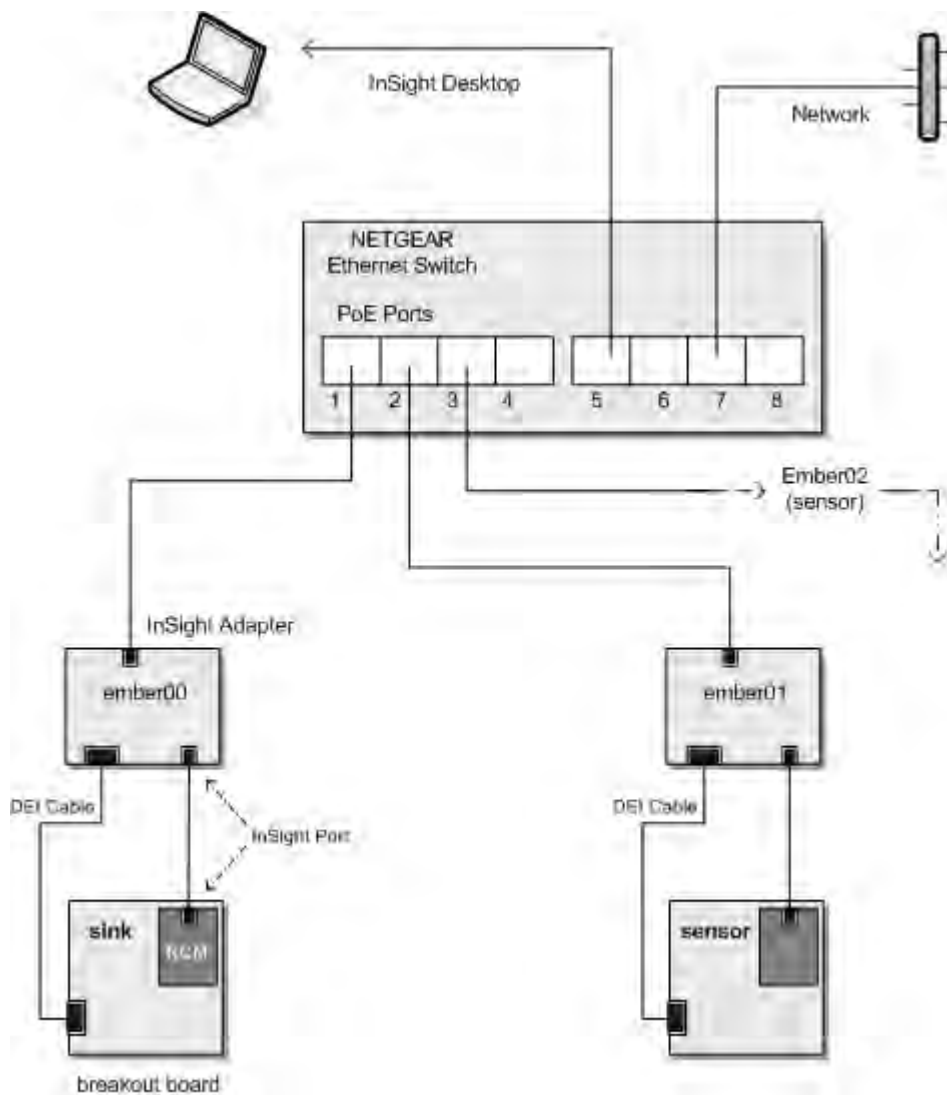


Figure 5. Setup of InSight development environment: one sink node and two sensors

## InSight Adapter Settings

As shipped, InSight Adapters are configured with DHCP mode enabled and a preassigned hostname is assigned:

- ember00 ...ember07 (EM260 Development Kit)
- ember00 ... ember02 (JumpStart Edition for the EM260)

Each InSight Adapter carries a label with its hostname. Hostnames for all InSight Adapters in the Development Kit also appear on a printed hardware ID list, which are mapped to their EUI-64 identifiers, and the serial numbers of matching radio communication modules.

**Caution:** InSight Adapters are accessible through their assigned hostnames only if your DHCP system populates DNS records automatically. Otherwise, you must access them through their IP addresses. Alternatively, you can set up a hosts file on your local computer that maps host names to IP addresses. If you are using static IP addresses, make sure that they do not conflict with other network addresses.

## Hardware Connections

To avoid communication problems, verify that each radio communication module is firmly seated in its breakout board connector, and its InSight port cable is properly seated.

## EUI-64 Assignments

Each radio communication module in the Development Kit has a serial number which you can use to identify its IEEE EUI-64 assignment. The serial number to a radio communication module is printed on a label affixed to the back of the module. To see the label, remove the module from the breakout board. Your Development Kit includes a hardware ID list which maps each module serial number to its assigned EUI-64.

## Attaching and Removing Breakout Board Modules

Two hardware modules attach to the EM260 breakout board:

- Radio communication module
- InSight Adapter

**Caution:** Observe electrostatic discharge (ESD) precautions when removing, handling, or replacing RF communication modules.

### To attach the radio communication module:

1. Align the radio communication module to its footprint outline on the breakout board
2. Press it into place, making sure that the radio communication module sits firmly in its mating connector.

### To remove the radio communication module:

- Carefully pull the module away from the breakout board.

### To attach the InSight Adapter:

1. Plug the InSight port cable into the InSight Adapter.
2. Plug the InSight port cable into the radio communication module.
3. Verify that the InSight Adapter's RCM Power Select switch is set to *Int* (Internal).

### To remove the InSight Adapter:

1. Unplug the InSight port cable from the radio communication module.
2. Unplug the InSight port cable from the InSight Adapter.

## Powering the Breakout Board

You can power the breakout board from an InSight Adapter, a USB cable connected to a PC, or directly from the supplied AC adapter.

### Powering the breakout board from an InSight Adapter

An InSight Adapter obtains its power from the supplied AC adapter (Figure 3) or the PoE switch ("Power-Over-Ethernet (PoE) switch"). In either case, you can set up the breakout board and its radio communication module to obtain power from the InSight Adapter as follows:

1. Make sure that the RCM Power Select switch on the front of the InSight Adapter is set to the `Int` position.
2. Plug one end of the supplied InSight Port cable into the InSight Adapter's InSight Port interface.
3. Plug the other end of the InSight Port cable into the radio communication module's InSight Port connector.

## Powering the breakout board from a USB cable connected to a PC

1. Connect the USB cable to the PC.
2. Plug the end of the USB cable into the USB jack on the board.

## Powering the breakout board directly

1. Snap an input plug into the developer kit's AC power adapter.
2. Plug the power adapter into a wall outlet.
3. Plug the power adapter into the 12V power jack on the board.

## Resetting InSight Adapters

To perform a hardware reset of an InSight Adapter, use a pen or pencil point to press the red reset button on the front panel.

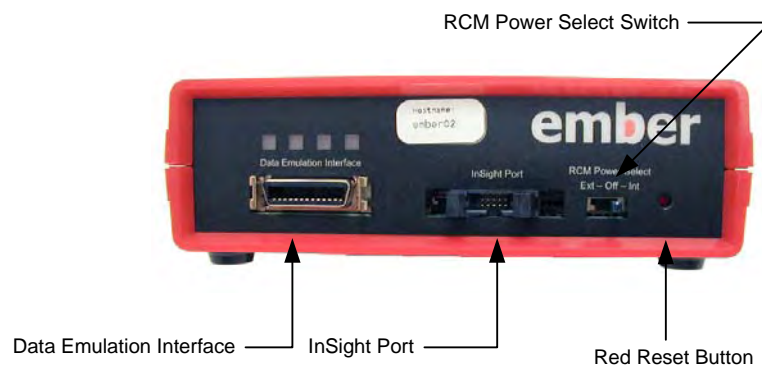


Figure 6. InSight Adapter Front Panel

## Connecting the SMA Connector to the RCM

The Development Kit contains a cable that allows SMA access to the EM260's radio frequency signal. Plug this cable into the Murata RF switch connector.

## Battery Powering Modules

**Caution:** Observe electrostatic discharge (ESD) precautions when handling RF communication modules that have been removed from the breakout board.

To increase deployment options for your application, use the supplied battery pack (with two AAA batteries) to power the radio communication module via the breakout board. You must connect the module to the breakout board. To locate the battery pack connector, see Figure 1.

**To connect the battery pack:**

1. Insert two AAA batteries into the battery pack.
2. Observing ESD precautions, remove the radio communication module from the breakout board.
3. Plug the battery pack into the battery pack port (J4) on the breakout board.

**Configuring the Serial Communication Path**

You can directly access a host microcontroller UART from either the breakout board USB connection or by telnetting into port 4901 of its InSight Adapter. For detailed information on all methods, see the *EM260 Breakout Board Technical Specification: SPI/UART, 128kB Version (120-2008-000)*.

To access the host microcontroller UART from port 4901 of the InSight Adapter, configure the serial communication path as follows:

1. Disconnect the InSight data emulation interface (DEI) cable from the InSight Adapter.
2. Configure switches SW1 (SPI, pin 1, left) and SW2 (DEI, pin 2, right) and connect jumpers at J15, J18, J19, J20, J31, and J35, as shown in Figure 7.

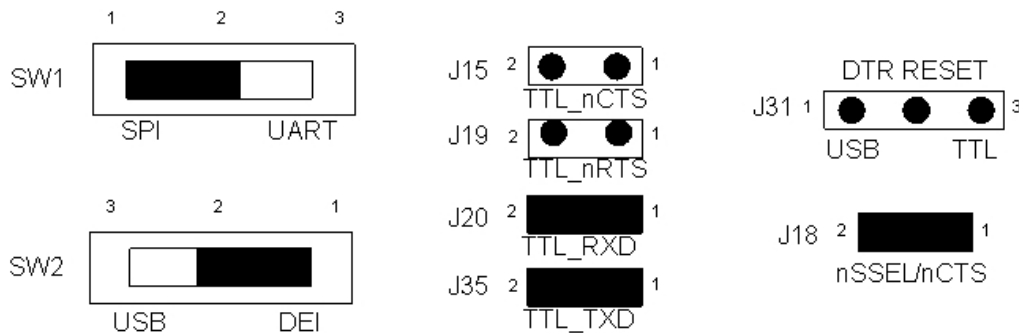


Figure 7. Switch and jumper settings required for host microcontroller UART access by InSight Adapter

3. Move the InSight Adapter voltage toggle switch to the `INT` position.
4. Remove the secondary power source from the EM260 breakout board.

**Caution:** Because the InSight data emulation interface is not buffered, you must provide power to the EM260 breakout board only from its InSight Adapter.

5. Connect the InSight DEI cable to the InSight Adapter.

This completes the hardware setup.

---

# 5

## Development Kit Components

With the Development Kit, you typically set up an InSight network that you use while developing and debugging applications. All nodes in this InSight network are connected to the Ethernet via their InSight Adapters. An Ethernet-connected workstation that is running InSight Desktop uses the InSight Adapters to monitor network node data and manage node behavior.

To set up an InSight network:

1. Install InSight Desktop on a computer that is connected to your network.
2. Connect all InSight Adapters to the Ethernet.
3. Power up all nodes (see “Powering the Breakout Board”).
4. **(Optional)** In order to establish a serial port connection to a node so that you can interact with the node’s application:
  - Telnet to the serial port 4902 (UART1).
  - Issue the InSight Adapter command `port 1` to check the settings for UART1.
  - Configure the InSight Adapter serial port settings to match the application settings. For example: `port 1 115200 8-n-1`
  - Quit the Telnet session.

For detailed information, see “Configuring the Serial Communication Path.”
5. Distribute nodes in the test environment, following the guidelines in “Positioning Network Nodes.”

### Positioning Network Nodes

You can position radio communication modules in any spatial orientation (for example, horizontally, vertically, or upside-down), but for optimal performance all should be positioned with their antennas in the same orientation. Observe the operating temperature limits described in the technical specification for the radio communication module in your Development Kit.

In general, objects near modules do not significantly affect radio transmission. Very large solid objects, such as walls and metal objects, can diminish the radio range of modules positioned within a few inches of them. Placing a module within a metal box can completely isolate the module from the network. Repositioning the module 20 to 30 cm from any obstructions will usually restore full range. You can also reconfigure the network to work around a radio obstacle.

Other IEEE 802.15.4 networks may compete for available bandwidth. Changing channels usually takes care of any conflicts.

---

# 6

## Using the InSight Adapter

The Development Kit InSight Adapter provides an Ethernet connection to networked radio communication modules and to development tools on the breakout board—for example, programmable buttons. You can use the InSight Adapter to perform the following tasks:

- Debug hardware.
- Monitor application or debugging data over radio communication module serial ports.
- Monitor radio data packets.

For detailed information on InSight Adapter commands, see Chapter 7, “InSight Adapter Commands.”

### InSight Adapter Default Configuration

InSight Adapters are shipped in the following state:

- DHCP mode with pre-assigned hostnames:
  - `ember00 ...ember07` (EM260 Development Kit)
  - `ember00 ... ember02` (JumpStart Edition for the EM260)
- Serial port settings of 115200 baud and 8-n-1

You can reconfigure the modules to use static mode (see “Changing to static IP address mode”).

### Connecting to an InSight Adapter Socket

You can telnet via Ethernet into the InSight Adapter sockets listed in Table 6, using either the hostname or the IP address of the InSight Adapter (your network must be configured appropriately for hostname access).

**Caution:** The administrative port (4902) is also available from the standard telnet port (23).

Table 6. InSight Adapter sockets

Socket	Definition	Use
4900	Radio communication module virtual serial port.	Monitor output for application development or serial port debugging. These ports also let users simulate external input.
4901	Radio communication module serial port 1, accessible only via a connection to the InSight data emulation interface.	
4902	Administrative	InSight Adapter command line interface
4903	InSight Adapter socket reset	Reset the sockets, but does not reset the adapter itself. Telnet to 4903 and issue the reset command.
4911	Radio communication module serial port 1	Monitors input for application development or serial port debugging.

## Monitoring Radio Communication Module Data

You can monitor or simulate real-time traffic between a radio communication module and external devices via either of the module's serial ports. You can drive these serial ports from Ethernet sockets 4900 or 4901, from the breakout board serial ports, or from the TTL header. To do this, you must configure port settings for the breakout board serial ports to match those being used by the application running on the radio communication module.

**Note:** Socket 4901 is only accessible over an InSight data emulation interface connection.

### To configure a breakout board serial port for radio communication module monitoring:

Issue the InSight Adapter command `port` and specify the port settings that are used by the application running on the radio communication module. For example, if your application uses breakout board serial port 1 and settings 115200 8-n-1, issue this command:

```
port 1 115200 8-n-1
```

## Connecting to the USB Administrative Port

To issue commands to the InSight Adapter through a USB port:

1. Connect a USB cable from your workstation to the InSight Adapter's USB connector.
2. From your workstation, run a standard terminal emulator (for example, Hyperterm, TeraTerm, PUTTY) to connect to the InSight Adapter (the InSight Adapter appears as a COM port).
3. Configure your terminal emulator with these settings:
  - 115200 (baud)
  - 8 (data bits)
  - n (no parity)
  - 1 (stop bit)
  - Flow Control = None

## Resetting an InSight Adapter Socket

If an Ethernet socket become inaccessible, issue the `reset` command on the InSight Adapter administrative socket 4903.

## Resetting the InSight Adapter

If you cannot connect to socket 4903, the InSight Adapter administrative socket, or if the `reset` command does not work, you can force a reset in one of the following ways:

- Press the InSight Adapter's red RESET button in front.
- If this method fails, power cycle the breakout board by disconnecting the external power supply or the PoE cable.

## Terminating a Connection to the Administrative Socket

To terminate a connection to the administrative socket, issue the `quit` command.

## Configuring the InSight Adapter

You can configure the InSight Adapter in several ways:

- Change from DHCP to static IP address mode.
- Change the host name.
- Configure serial port settings.

You can also view current configuration settings by issuing the `config` command via telnet to the InSight Adapter administrative socket 4902. This command returns all non-volatile configuration information for the InSight Adapter:

- hostname
- serial port information
- static IP settings
- date and time of the last compile of the module's software
- whether DHCP is in effect

If DHCP mode is in effect, the command also outputs its current dynamic IP settings.

To determine the version of the InSight Adapter firmware, issue the `version` command on the InSight Adapter administrative socket 4903.

## Changing to static IP address mode

InSight Adapter is preconfigured for DHCP. You can change this configuration to implement static IP address mode.

**To configure the Ethernet connection for static mode using the USB administrative port:**

1. Set up your USB administrative connection as described in "Connecting to the USB Administrative Port".
2. Issue the following InSight Adapter commands through the terminal emulator:
 

```
ip_static <ipaddress> <netmask> <gateway>
ip_dhcp off
```
3. Reboot the InSight Adapter by pressing the red RESET button on the front of the InSight Adapter.

To configure the Ethernet connection for static mode using the telnet administrative port:

1. Telnet into socket 4902.
2. Issue the following InSight Adapter commands through the terminal emulator:
 

```
ip_static <ipaddress> <netmask> <gateway>
ip_dhcp off
```
3. Reboot the InSight Adapter by pressing the red RESET button on the front of the InSight Adapter.

## Changing the hostname

If you purchase two developer kits, you might need to change the hostnames assigned to the InSight Adapters in one of the kits in order to avoid network conflicts. You can change a hostname by issuing the `hostname` command to the InSight Adapter in two ways:

**To change the hostname via an InSight Adapter USB connection:**

1. Set up your USB administrative connection as described in “Connecting to the USB Administrative Port”.
2. Issue the InSight adapter command `hostname` through the terminal emulator, giving it the new hostname. For example:
 

```
hostname set <new-hostname>
```
3. To verify the new host name, issue the InSight Adapter command `config`, which returns with the new hostname.

**To change the hostname using a telnet administrative port:**

- Telnet into socket 4902.

## Configuring serial port settings

To change the breakout board serial port settings, connect to the InSight Adapter administrative socket (4902) and issue the `port` command. For example, the following command configures breakout board serial port 1 for 19200 baud rate, 8 bits, no parity, and 1 stop bit:

```
port 1 19200 8-n-1
```

## Getting Help on InSight Adapter Commands

To view a description of an InSight Adapter command, issue the `help` command followed by a command name—for example, `help port`.

## Uploading Software to the InSight Adapter

Periodically, Ember provides software upgrades for the InSight Adapter. You can upload these upgrades using InSight Desktop. To do this, right click on the adapter in the Adapter View window and select Upload Adapter Firmware.

# 7

## InSight Adapter Commands

### Syntax Conventions

Element	Meaning	Example
literal	Indicates a syntax element—command or argument—that is entered exactly as shown.	<code>ip_dhcp {on off}</code>
variable	Indicates a variable name.	<code>help [command-name]</code>
{ }	Delimits a set of one or more mandatory arguments	<code>{a}</code> <code>{a b}</code>
[ ]	Delimits a set of one or more optional arguments	<code>[0 1]</code> <code>[rate]</code>
	Delimits options within a list, where only one can be chosen.	<code>{a b c}</code> <code>[red blue green yellow]</code>

### Entering Commands

- Commands are not case sensitive.
- Arguments are separated with a single white space.
- To move the cursor on the command line, press the arrow-left or arrow-right key.
- Press CTRL-D to delete all characters to the right of the cursor.
- Press BACKSPACE to delete one character to the left of the cursor.

**Note:** The interface times out after 10 minutes of inactivity.

### Connecting to the InSight Adapter Administrative Port

To connect to the InSight Adapter administrative port, do one of the following:

- Telnet into port 4902 over the Ethernet.

### InSight Adapter Commands

- `button`
- `config`
- `gpio`
- `help`

- hostname
- ip\_dhcp
- ip\_static
- port
- power
- quit
- reset
- socket\_reset
- upload\_backchannel
- version

---

## button

```
button {0|1} {on|off}
```

### Description

Sets the specified breakout board button to a state of on or off.

You can issue this command only over an InSight data emulation interface connection.

**Warning:** In order to use the InSight data emulation interface, you must power the radio communication module from an InSight Adapter that is powered by its Ethernet connection, or from the development kit's 12V DC power supply. Powering the radio communication module from the battery pack when the InSight data emulation interface is in use can damage the InSight Adapter.

### Arguments

0, 1	Button number, where 0 maps to breakout board button EM1, and 1 maps to EM2.
on, off	The button setting

### Example

```
button 1 on
```

---

## config

```
config
```

### Description

Outputs all non-volatile configuration information for the InSight Adapter:

- hostname
- static IP settings
- whether DHCP mode is in effect
- serial port information
- date and time of the last compile of the module's software (needed to determine the software version).

If DHCP mode is in effect, the command also outputs its current dynamic IP settings.

You can also view configuration information via InSight Desktop (see the *InSight Desktop User's Guide*).

**Arguments**

None.

---

**gpio**`gpio``gpio {read | monitor}``gpio {config | set} 0xNNNN`**Description**

Allows reading and writing of GPIO pin states to the attached EM250 or ATmega host MCU. (Only applies to pins configured as GPIO, with no alternate functions.)

If issued with no arguments, reports on config, set, and read. Bits 19:17, 10:9, and 5:4 are 0.

Parameter `0xNNNN` corresponds to a bitmask, specified in big-endian notation, indicating GPIO bits 16-0.

**Arguments**

<code>set</code>	Report or configure output level high (1) or low (0). Bits 19:17, 10:9, and 5:4 are ignored.
<code>monitor</code>	Report on input changes sampled at 0.1 seconds till key press. Bits 19:17, 10:9, and 5:4 are 0.
<code>read</code>	Report on input. Bits 19:17, 10:9, and 5:4 are 0.
<code>config</code>	Same as <code>set</code>

**Example**`gpio read`

---

**help**`help [ command-name ]`**Description**

Provides help on InSight Adapter commands as follows:

- If issued with no arguments, outputs a list of all InSight Adapter commands.
- If issued with a valid command name, outputs a description of that command.

**Arguments**

The name of any InSight Adapter command.

**Example**`help port`

---

**hostname**`hostname set hostname``hostname read`

**Description**

Valid only for DHCP mode, use this command to set or obtain the host name of the InSight Adapter. The module tries to set this hostname when it connects to the Ethernet. The hostname that you set is used only after you reboot the module.

You can also view the hostname via InSight Desktop (see the *InSight Desktop User's Guide*).

**Arguments**

<code>set</code>	Specifies that the command identify a host name
<code>hostname</code>	Character hostname, may include up to 20 alphanumeric characters
<code>read</code>	Specifies that the command outputs a hostname

**Example**

```
hostname set myhost1
```

**ip\_dhcp**

```
ip_dhcp {on|off}
```

**Arguments**

<code>on</code>	Specifies to use DHCP.
<code>off</code>	Specifies not to use DHCP.

**Description**

Configures the InSight Adapter to use DHCP. This command is a prerequisite for using the hostname command.

For the command to take effect, you must reboot the InSight Adapter.

You can power cycle a module via InSight Desktop (see the *InSight Desktop User's Guide*).

**Example**

```
ip_dhcp on
```

**ip\_static**

```
ip_static ipaddress netmask gateway
```

**Description**

Configures the InSight Adapter statically. This command takes effect only after you reboot the InSight Adapter.

**Arguments**

<code>ipaddress</code>	IP address of the breakout board
<code>netmask</code>	Netmask for current network
<code>gateway</code>	Gateway for your network configuration

**Example**

```
ip_static 192.168.168.126 255.255.255.0 192.168.168.1
```

## port

```
port {0|1} [ baud-rate [ #bits-parity-stopbits ] ]
```

### Description

Configures serial ports on a breakout board. If you supply only the serial port number, the port command returns with that port's current configuration.

Configures serial port settings for the breakout board. You can configure just the rate or the rate plus the data settings, but you cannot configure the data settings alone.

### Arguments

0, 1	The emulator/debug module's serial port number
<i>baud-rate</i>	The baud rate for this port.
<i>#bits</i>	Number of data bits per character
<i>-parity</i>	Parity with one of the following characters: <ul style="list-style-type: none"> <li>▪ N (none)</li> <li>▪ O (odd)</li> <li>▪ P (even)</li> </ul>
<i>-stopbits</i>	Stop bits with an integer of 1 or 2.

### Examples

To configure serial port 1 for a baud rate of 115200, 8 bits, no parity, 1 stop bit:

```
port 1 115200 8-n-1
```

To configure serial port 1 for a baud rate of 57600:

```
port 1 57600
```

To obtain the current configuration information for serial port 1:

```
port 1
```

## power

```
power {on|off|read}
```

### Description

Controls the power supplied to a radio communication module.

You can also power cycle a module via InSight Desktop (see the *InSight Desktop User's Guide*).

### Arguments

on	Powers the module on
off	Powers the module off
read	Outputs the status of radio communication module (on or off)

### Examples

```
power on
```

```
power read
```

**quit**

quit

**Description**

Terminates a connection to the administrative socket.

**Note:** This command has no effect on serial port connections to the admin port.

**Arguments**

None.

**reset**

reset

**Description**

Resets a radio communication module by pulling down its reset line for a millisecond.

You can also reset a module via InSight Desktop (see the *InSight Desktop User's Guide*).

**Arguments**

None.

**socket\_reset**

socket\_reset {4900|4901|4911}

**Description**

Forces a socket on the InSight Adapter to close. Use this command when port 4900, 4901, or 4911 becomes inaccessible because the socket closed incorrectly. (For port definitions, see Table 6.)

**Note:** Port 4901 is only accessible over an InSight data emulation interface connection.

**Arguments**

4900	Serial port 0
4901	Serial port 1
4911	Serial port sniff

**upload\_backchannel**

upload\_backchannel

**Description**

Puts the InSight Adapter into FTP server mode to enable software uploads.

**Arguments**

None.

---

## **version**

version

### **Description**

Outputs the software version number of the InSight Adapter software.

You can also view the version number via InSight Desktop (see the *InSight Desktop User's Guide*).

### **Arguments**

None.