RN-BT-AUDIO-UG

Bluetooth Audio Module
Command Reference User's Guide

MODULES SUPPORTED:
RN52-I/RM

RN-BT-AUDIO-UG Version 1.0 1/28/13

Roving Networks, Inc.
102 Cooper Court
Los Gatos, CA 95032
+1 (408) 395-5300
www.rovingnetworks.com

Table of Contents

Chapter 1. Introduction

1.1 Overview 5
1.2 Evaluation Boards & Reference Designs 5
1.3 Bluetooth Profiles 6
1.4 Command Mode vs. Data Mode 6
1.5 Using GPIO Pins for Configuration 8
1.6 Making a Bluetooth Connection 9
1.7 Security Modes 10

Chapter 2. Command Reference

2.1 Command Syntax 13
2.2 Set Commands 14
2.3 Get Commands 17
2.4 Action Commands 18
2.5 Audio Commands 21

Appendix A. Command Quick Reference Guide

Appendix B. Firmware Revision History

B.1 Version 1.05 25

Appendix C. Document Information

Chapter 1. Introduction

1.1 OVERVIEW

This document contains the software command reference and advanced configuration settings for Roving Networks Bluetooth audio devices. Commands and settings that are specific to a single product or product family are identified as such in the document.

You program Roving Networks Bluetooth devices through the UART interface using a simple ASCII command language. Set commands configure the module and get commands echo the current configuration. Configuration settings modified with the set command do not take effect until the module has been rebooted, even though the get command may show otherwise.

This document assumes that you have a working knowledge of Bluetooth operation and communications. To program the Roving Networks devices you need a Bluetooth-enabled PC (either built-in or using a USB Bluetooth dongle). You can only configure one device at a time. Once configured, device settings are saved (independent of power down) until they are explicitly changed or the factory defaults are restored.

NOTICE TO CUSTOMERS

The commands and applications described in this document apply to Roving Networks Bluetooth *audio modules*, e.g., the RN52. They do not apply to Roving Networks Bluetooth data modules such as the RN41, or RN42. For data module configuration information, refer to the *Bluetooth Data Module Command Reference & Advanced Information User's Guide*.

1.2 EVALUATION BOARDS & REFERENCE DESIGNS

Roving Networks provides a variety of boards, kits, and reference designs that you can use for evaluation and prototyping.

The RN-52-EK evaluation board is Bluetooth SIG qualified prototyping platform for the RN52 module. The board has the flexibility to connect directly to PCs via a standard USB interface (via the FTDI chipset) or to embedded processors through the TTL UART interface. The status LEDs, switches, and signal headers enable rapid prototyping and integration into existing systems.

For more information on available evaluation boards and reference designs, refer to the Roving Networks web site.

1.3 BLUETOOTH PROFILES

Roving Networks audio modules support several Bluetooth profiles, as described in Table 1-1. The default profile is the serial port profile (SPP). You set the profile using ASCII commands when the module is in command mode.

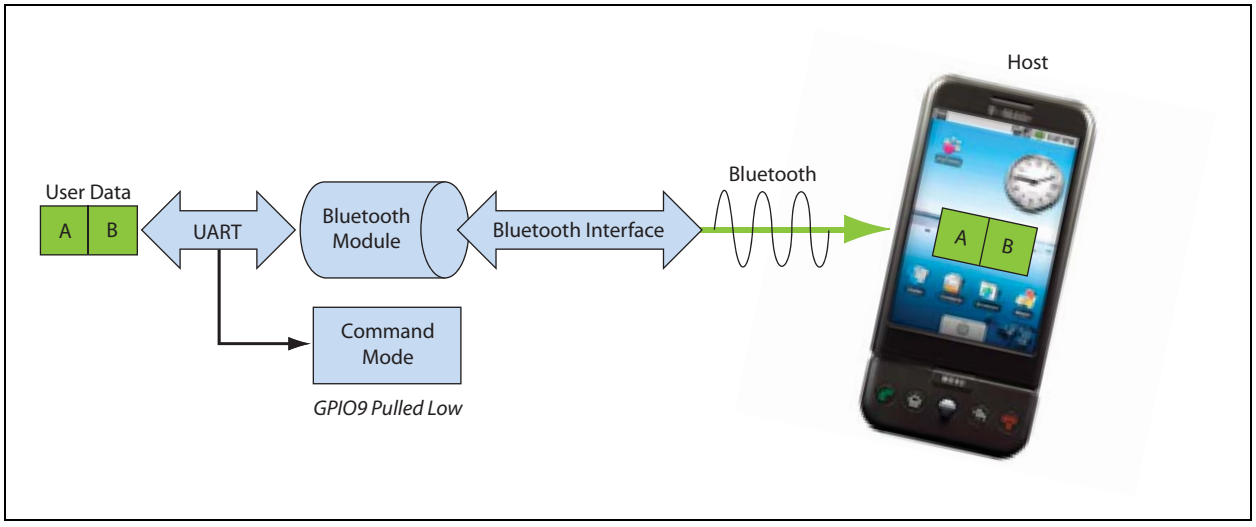
TABLE 1-1: SUPPORTED BLUETOOTH PROFILES

Profile	Comments
SPP	SPP defines how to set up virtual serial ports and connect two Bluetooth enabled devices. SPP emulates a serial cable link over Bluetooth wireless technology.
A2DP	The advanced audio distribution profile (A2DP) defines how high quality audio (stereo or mono) can be streamed from one device to another over a Bluetooth connection.
HFP	The hands free profile (HFP) is commonly used to allow car hands-free kits to communicate with mobile phones in the vehicle.
iAP	The module natively supports iPod Accessory Protocol (iAP) data connections and directly manages authentication, reducing engineering effort and cost, and simplifying accessory product design.

1.4 COMMAND MODE VS. DATA MODE

The Bluetooth device operates in two modes: data mode (default) and command mode. While in data mode, the module is essentially a data pipe. When the module receives data, it strips the Bluetooth headers and trailers and passes the user data to the UART. When data is written to the UART, the module constructs the Bluetooth packet and sends it out over the Bluetooth connection. Thus, the entire process of sending/receiving data to the host is transparent to the end microprocessor. See Figure 1-1.

FIGURE 1-1: DATA & COMMAND MODES



You configure the device by putting it into command mode and sending ASCII commands over a serial (UART) port. Once you change the configuration parameters, they persist until you change them or perform a factory reset.

NOTICE

You can only configure the Bluetooth audio module locally using your computer's serial port. You *cannot* configure the module remotely over the Bluetooth link.

1.4.1 Default Configuration & Serial Port Settings

Table 1-2 shows the default configuration for the Bluetooth module:

TABLE 1-2: DEFAULT CONFIGURATION & SERIAL PORT SETTINGS

Option	Setting
Bluetooth mode	Slave
Bluetooth pin code (for legacy pairing modes)	1234
Baud rate	115,200 Kbps
Bits	8
Parity	None
Stop bits	1
Flow control	Disabled

1.4.2 Configuring the Module over the UART

Connect the evaluation board to your computer using a USB cable. With the Bluetooth module connected and powered on, run a terminal emulator and open the COM port to which the cable is connected. The terminal emulator's communication settings should be the default serial port settings of the Bluetooth module.

Note: You can use local configuration at any time when the device does NOT have a Bluetooth connection, as well as under certain conditions. If the device is in configuration mode and a connection occurs, the device exits configuration mode and data passes back and forth from the remote device.

When you are finished configuring, reset the device, which causes the device to exit configuration mode and allows data to pass normally.

1.4.3 Enter Command Mode

Launch a terminal emulator and specify the module's default settings (see Table 1-2).

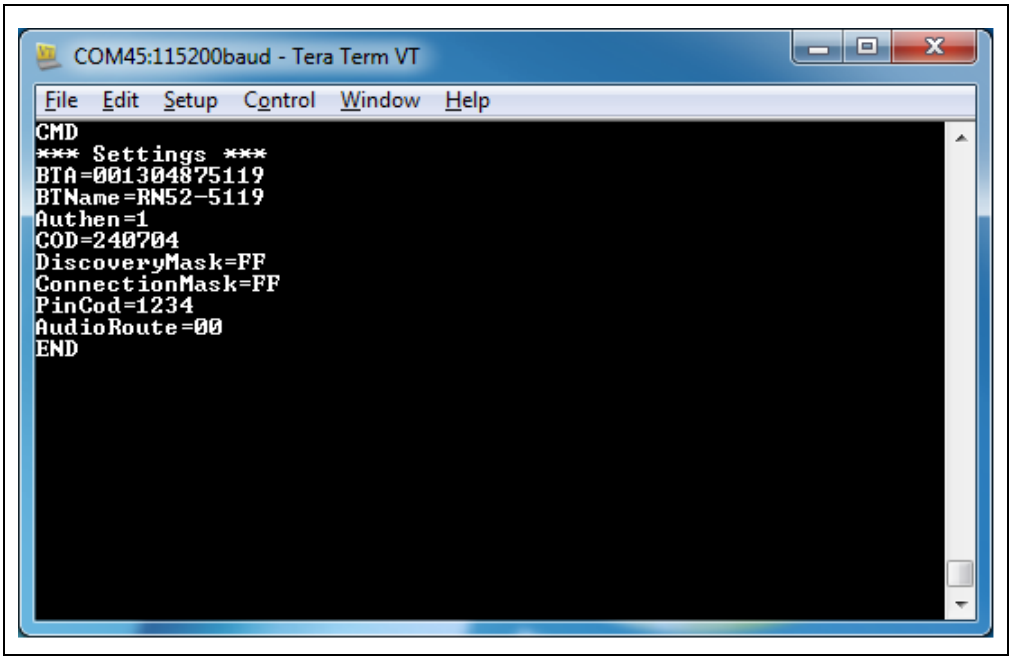
Drive GPIO9 low to enter command mode. The module returns the string `CMD` to the UART console to indicate that the module is in command mode. While in command mode, the device accepts ASCII bytes as commands.

When you enter a valid command, the module returns `AOK` . It returns `ERR` for an invalid command and `?` for unrecognized commands. Type `h` <cr> to see a list of commands.

A quick check to confirm that you are in command mode is to type the `D` <cr> command after entering command mode. This command shows the a summary of the module's current settings, such as the Bluetooth name, device class, and serial port settings. See Figure 1-2.

To return to data mode, drive GPIO9 high or reset the device and re-connect. When leaving command mode the module sends END to the UART.

FIGURE 1-2: VIEW CURRENT SETTINGS



1.5 USING GPIO PINS FOR CONFIGURATION

Roving Networks Bluetooth modules have GPIO pins that you can use to configure the module. See Table 1-3.

TABLE 1-3: GPIO SETTINGS

GPIO Pin	Function	Settings (OFF = 0 VDC/ON = 3 VDC)
GPIO2	Profile Change	Toggles low for 100 ms when the module detects a profile connection status change.
GPIO4	Factory Reset	Off = disabled, on = armed. Set this GPIO pin on power up to arm the reset function. Then toggle the device off and on three times to reset all settings to the factory defaults (other than the Bluetooth name).
GPIO7	Baud Rate	Off = 115 K, on = 9,600. This setting is used to configure 9,600 or a software selected (default = 115 K) baud rate. If the GPIO signal is low, the device uses the stored baud rate setting. When the GPIO pin is high, the baud rate is set to 9,600 regardless of the software setting.
GPIO9	Command Mode	Module enters command mode on the falling edge of GPIO9. Module leaves command mode on the rising edge of GPIO9.

The module can drive status LEDs that give you a visual confirmation that the module or board is powered up and operating. Table 1-4 describes the status LEDs.

TABLE 1-4: STATUS LED FUNCTIONS

LED	Status	Description
LED0 and LED1	Flashing	The RN52 module is discoverable.
LED0 only	Flashing	The module is connected.
LED1 only	Flashing	The module is connectable.

1.6 MAKING A BLUETOOTH CONNECTION

By default, the Bluetooth module acts as a slave and the PC or host MCU is the master. You connect to the Bluetooth module using your computer's Bluetooth device manager, which varies depending on the operating system. Regardless of the operating system, the process is the same: discovery, pairing, and connecting.

1.6.1 Discovery

Upon power up, the module is discoverable. Open your PC's Bluetooth device manager and choose to add a new device. The Bluetooth device manager's icon is located in the bottom right corner of your screen in the taskbar for Windows and in the upper right corner for Mac OS-X. The Bluetooth device manager displays a list of discoverable Bluetooth devices. The module displays as **RN52-***XXXX*, where *XXXX* is the last 4 digits of the module's MAC address. The module's label also shows the MAC address.

1.6.2 Pairing

To pair with the module, double-click its name in the list. The module's firmware automatically stores up to 8 pairings from remote hosts in a first in, first out fashion.

The default authentication mode is keyboard (no pin code required). When the Bluetooth device manager completes pairing, it issues a message that the Bluetooth device is installed on COM*X* where COM*X* is unique to your computer. In some cases, the Bluetooth device manager creates two COM ports; in this situation, only use the COM port labeled "outgoing."

If the remote Bluetooth device does not require authentication, a connection can occur without the pairing process. However the Bluetooth specification requires that if either device involved in the pairing process requires authentication, the other device must participate to ensure a secure link. Roving Networks modules default to simple secure pairing (SPP) open or keyboard I/O mode and do NOT require authentication. However, most PCs do not support this mode and, therefore, require authentication. In this case, use the module's default pin code, 1234, as the pass key. See "Security Modes" on page 10 for more information on using pass keys.

The module may use SSP when it attempts to pair with devices that support the Bluetooth specification version 2.1 + EDR. SSP does not require the user to remember the pin code, but it asks to confirm a 6-digit number if the device has a display capability.

Once connected, the module is in data mode, allowing data to flow in both directions. For configuration, the device must be in command mode. See “Enter Command Mode” on page 7 for more information.

Note:	Only one client can connect to a slave device at a time. As a master, the device can make multiple connections, but only in a point-to-point, serialized manner. Roving Networks modules do not currently support multi-point master mode.

Figure 1-3 shows some pairing/connecting examples on several platforms.

FIGURE 1-3: PAIRING/CONNECTING WITH THE BLUETOOTH ADAPTER

1.6.3 Connecting

To establish a Bluetooth connection, open the module’s COM port from your application or a terminal emulator. The module remains connected until you close the COM port or remove power from it.

1.7 SECURITY MODES

The Bluetooth module supports authentication. If the local or remote Bluetooth device has authentication enabled, you must enter a pin code the first time you attempt to connect. The pin code is a series of alphanumeric characters from 1 to 16 characters in length. The default pin code is 1234.

After you enter the pin code, the Bluetooth devices compare them. If they match, a link key is generated and stored. Usually, but not always, the remote device stores the link key. For subsequent connections, the devices compare link keys. If they are correct, you do not need to re-enter the pin code.

If the remote device is a PC or PDA, the user generally is prompted to enter this pin code. To remove the stored link key on the remote device, you typically “unpair” or remove the device from the Bluetooth manager. You can change the pin code to remove the link key on the Bluetooth adapter, forcing a new pin code exchange to occur upon subsequent connection attempts.

Note: Only one master can connect to the Bluetooth module at a time.

NOTES:

Chapter 2. Command Reference

Roving Networks Bluetooth modules support a variety of commands for configuration. This section describes these commands in detail and provides examples.

NOTICE TO CUSTOMERS

The commands and applications described in this document apply to Roving Networks Bluetooth *audio modules*, e.g., the RN52. They do not apply to Roving Networks Bluetooth data modules such as the RN41 or RN42. For data module configuration information, refer to the *Bluetooth Data Module Command Reference & Advanced Information User's Guide*.

2.1 COMMAND SYNTAX

To issue commands to the module, you send a keyword followed by optional parameters via the UART.

- All commands are one or two characters and can be upper or lower case.
- Delimit command arguments with a comma.
- Commands use decimal input, except where noted.
- Text data, such as the Bluetooth name and pin code, is case sensitive.
- Set commands only take effect AFTER reboot, except where noted.

There are five general command categories, as shown in Table 2-1.

TABLE 2-1: COMMAND TYPES

Command Type	Description
Set commands	Store information to flash memory. Changes take effect after a power cycle or reboot.
Get commands	Retrieve and display the stored information.
Action commands	Perform actions such as inquiries, connecting, etc.
Audio commands	Control the audio playback.

Each command terminates with the carriage return character (\r, decimal 13, hex 0x0D).

2.2 SET COMMANDS

The set commands specify configuration settings and take effect after power cycling or rebooting.

2.2.1 S|,<*hex value*>

This command sets the routing for the audio output, where <*hex value*> is an 8-bit hex value shown in Table 2-2.

TABLE 2-2: AUDIO ROUTING VALUES

Value	Description
00	Analog output (default).
01	Set the output for I^2S. See Table 2-3.
02	Set the output for S/PDIF.

TABLE 2-3: RN52 I^2S SETTINGS

Parameter	Value
Configuration	Master mode
Supported sample rates	8, 11.025, 12, 16, 22.05, 24, 32, 44.1 and 48kHz
Sample width	24 bits
Synchronous data	Data is right channel with word select (WS) high. The SD data MSB occurs in the second SCLK period.
Justification	Left justified.

Example: `S|,02` // Set the audio output to S/PDIF

2.2.2 S-,<*string*>

This command sets the module's normalized name where <*string*> is a prefix. The module's name is set to <*string*>-*XXXX*, where *XXXX* is the last four digits of the module's MAC address.

This setting is useful for situations in which you want to set up multiple modules with simular but unique identifiers.

Default: RN52

Example: `S-,MCHIP` // Set module's name to MCHIP-*XXXX*

2.2.3 SA,<*value*>

The set authentication command forces authentication when a remote device attempts to connect, where <*value*> is a decimal value shown in Table 2-4. Regardless of this setting, if a remote device forces authentication, this device responds with the stored pin code. Once a remote device has exchanged pin codes with this device, a link key is stored for future use. The device stores up to 8 keys automatically and permanently in flash memory, in a first in, first out fashion.

TABLE 2-4: SET AUTHENTICATION VALUES

Value	Description
0	Open. Authentication is not required. The device accepts pin code mode.
1	SSP keyboard I/O mode (default). If this option is set, the remote host receives a prompt; reply yes to pair. Optional does not force this mode but accepts it if the host requires (e.g., Droid 3.3+). The host posts a message asking for confirmation; the module always responds yes.
2	SSP “just works” mode. You can use this mode with Droid devices if the application connects using unsecure mode (which was the default on Droid version 3.3). This mode also works with new PC stacks.
4	Pin code. Forces pin mode, which requires the host device to enter a pin code that matches the stored pin code.

Note: Modes 0 and 4 are legacy modes that do not support SSP (Bluetooth version 2.0).

Default: 1 // Keyboard mode

Example: `SA,4` // Set to pin code mode

2.2.4 SC,<*hex value*>

This command sets the service class field in the class of device (COD), where <*hex value*> is a 24-bit hex number. The service class consists of the most significant 11 bits in the COD. This command sets the MSW to create the 24-bit device class number. The inquiring device interprets the service class to determine the service. A complete listing of available Bluetooth service classes is referenced on the Bluetooth SIG web site.

The default COD, 240704, represents the following device:

- Service class: rendering, audio
- Major device class: wearable
- Minor device class: wrist watch

Default: 240704

Example: `SC,240710`

```
// Set service class to:
// Service class: rendering, audio
// Major device class: wearable
// Minor device class: helmet
```

2.2.5 SD,<*hex value*>

This command sets the discovery mask, where <*value*> is an 8-bit hex value representing the profiles enabled for discovery. The Bluetooth profiles are represented by an 8-bit hex value as shown in Table 2-5.

TABLE 2-5: BLUETOOTH PROFILE BITMASK VALUES

Bit Position	Profile
01	iAP
02	SPP
04	A2DP
08	HFP

Default: FF

Example: `SD,01` // Set the discovery profile to iAP

2.2.6 SF,1

This command sets all module parameters to the factory defaults. The changes do not take effect until you reboot the module.

Example: `SF,1` // Invoke factory defaults

2.2.7 SK,<*hex value*>

This command sets the connection mask where <*hex value*> is an 8-bit hex value of the profile enabled for connection. The Bluetooth profiles are represented by an 8-bit hex value as shown in Table 2-6.

TABLE 2-6: BLUETOOTH PROFILE BITMASK VALUES

Bit Position	Profile
01	iAP
02	SPP
04	A2DP
08	HFP

Default: FF

Example: `SK,08` // Set the connection profile to HFP

2.2.8 SN,<*string*>

This command sets the device name, where <*string*> is up to 20 alphanumeric characters.

Default: RN52-*XXXX*, where *XXXX* is the last 4 digits of the module's MAC address

Example: `SN,MyDevice` // Set the device name to “MyDevice”

2.2.9 SP,<*string*>

This command sets the security pin code, where <*string*> is up to 20 alphanumeric characters. Each time the device pairs successfully, it saves the Bluetooth address. The device can store up to eight addresses on a first in first out basis. Using this command also erases all stored pairings. You can use the same value that is already set.

You cannot erase the pin code, however, you can overwrite the default pin code.

Default: 1234

Example: `SP,0123` // Set pin code to 0123

2.3 GET COMMANDS

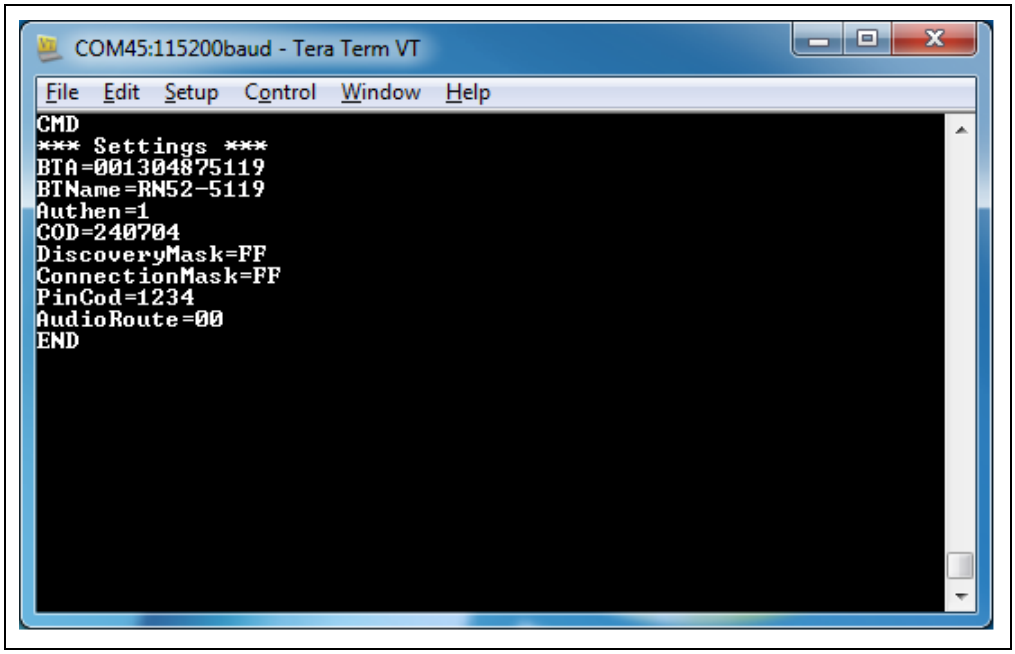
The get commands retrieve and display the device's stored information. These commands do not have a keyword or character and do not take any parameters, except as noted.

2.3.1 D

This command displays basic settings such as the address, name, UART settings, security, pin code, bonding, and remote address. Figure 2-1 shows an example of the output.

Example: `D` // Display basic settings

FIGURE 2-1: DISPLAY COMMAND EXAMPLE OUTPUT



2.3.2 G<*command*>

This command displays the stored settings for a set command, where <*command*> is a set command name.

Example: `GA` // Display the authentication mode

`GP` // Display the pin code

2.3.3 H

The help command displays a list of commands and their basic syntax.

Example: `H` // Display help

2.3.4 V

This command displays the firmware version.

Example: `V` // Show the firmware version

2.4 ACTION COMMANDS

Action commands perform actions such as inquiries, connecting, and entering/exiting command mode.

2.4.1 +

This command toggles the local echo on and off. If you send the `+` command in command mode, all typed characters are echoed to the output. Typing `+` a second time turns local echo off.

Default: Off

Example: `+` // Turn local echo on

2.4.2 @,<*flag*>

This command toggles whether the module is discoverable, where <*flag*> is 1 (discoverable) or 0 (not discoverable).

Example: `@,1` // Make the module discoverable

2.4.3 A,<*telephone number*>

This command initiates a voice call to a telephone, where <*telephone number*> is a telephone number up to 25 digits. The module returns an error (`ERR`) if the call status is not idle.

Example: `A,14083955300` // Call 1 (408) 395-5300

2.4.4 B

When you issue this command, the module attempts to reconnect the Bluetooth profiles specified in the connection mask to the most recently paired and connected device. See “SK,<hex value>” on page 16.

Use the `Q` command to retrieve the Bluetooth profile connection status in byte 0 (bits 0 - 3). The module returns an error if it has not been previously connected or if the connection mask is set to 00 (meaning the module is not connectable).

2.4.5 C

This command instructs the module to accept an incoming voice call. You use the `Q` command to retrieve the call status (bits 8 - 10) value. The module returns an error (`ERR`) if the call status is not set to incoming calls.

Example: `C` // Accept incoming call

2.4.6 E

This command terminates an active call or rejects an incoming call. The module returns an error (`ERR`) if the call stauts is not an incoming call or active call.

Example: `E` // Terminate call

2.4.7 F

This command forces a disconnect and terminates the Bluetooth link to the telephone. Discovery is not enabled when the Bluetooth link is dropped.

Example: `F` // Force disconnect

2.4.8 HV,<*value*>

This command sends a volume adjustment command to the telephone to adjust the voice call volume and synchronize the volume levels. <*value*> is the level in decimal integers from 0 - 15, and conforms to the HFP specification version 1.8 4.28.2.

Example: `HV,8` // Set volume to 8 level value

2.4.9 K,<*hex value*>

This command disconnects the currently active connection, where <*hex value*> is a bit-mask of the profile to disconnect**.** The characters `KILL`<cr><lf> are echoed to the local UART once the connection is broken. The Bluetooth profiles are represented by an 8-bit hex value as shown in Table 2-7.

TABLE 2-7: BLUETOOTH PROFILE BITMASK VALUES

Bit Position	Profile
01	iAP
02	SPP
04	A2DP
08	HFP

Example: `K,01` // Disconnect the iAP profile

2.4.10 M,<*flag*>

This command controls the hold/mute function for the current telephone call, where <*flag*> is 0 or 1. If <*flag*> is 1, the module mutes the call; if <*flag*> is 0, the call is unmuted.

Example: `M,1` // Mute the call

2.4.11 Q

This command queries the current connection status. It returns a multi-byte list of the bitmask value of the currectly connected profiles. Each value is expressed as 00 to FF. The byte list is terminated with the \r\n characters. Table 2-1 and Table 2-2 describes the byte 0 and byte 1 bit formats, respectively.

Example: The byte string `0C12\r\n` indicates that the A2DP and HFP profiles are connected, the call status is active, and that an audio mute/hold event was received from the phone.

Byte 1 event bits 2 - 4 are cleared when you issue the `Q` command. The status values do not change unless the module's status changes. The module drives GPIO2 low for 100 ms to notify attached equipment that the event/status register has been changed.

TABLE 2-1: BYTE 0 BIT FORMAT

Bit	Description
0	iAP wireless active connection to remote device.
1	SPP active connection to remote device.
2	A2DP active connection to remote device.
3	HFP/HSP active connection to remote device.
4 - 7	Reserved.

TABLE 2-2: BYTE 1 BIT FORMAT

Bit	Description
0 - 1	Call status. 0: idle, 1: incoming call, 2: active call.
2	HFP audio volume level change from audio gateway (phone). Use the `Y,0` command to retreive the volume level.
3	HFP audio microphone level change from audio gateway (phone). Use the `Y,1` command to retreive the volume level.
4 - 7	Reserved.

2.4.12 R,1

This command forces a complete device reboot (similar to a power cycle).

2.4.13 Y,<*flag*>

This command returns either the last speaker (<*flag*> = 0) or microphone level (<*flag*> = 1) as a 0 - 15 decimal value sent from the audio gateway in response to bits 14 and 15 in the event status register.

Example: `Y,0` // Returns the speaker volume

2.5 AUDIO COMMANDS

The audio commands control audio playback functions.

2.5.1 AV+

This command increases the volume.

2.5.2 AV-

This command reduces the volume.

2.5.3 AT+

This command plays the next track by sending an AVRCP volume previous track command to the host.

2.5.4 AT-

This command plays the previous track by sending an AVRCP volume previous track command to the host.

2.5.5 AP

This command pauses or starts playback by sending an AVRCP volume pause/play command to the host.

NOTES:

Appendix A. Command Quick Reference Guide

This section provides a quick reference of the firmware commands as well as the factory defaults. Table A-1 provides an overview of the set commands.

TABLE A-1: SET COMMANDS

Command	Description	Factory Settings
S|,<*hex value*>	Audio output routing.	00 (analog output)
S-,<*string*>	Sets the normalized name.	RN52
SA,<0,1,2,4>	Authentication enable/disable.	0, Disabled
SC,<*hex value*>	Service class.	240704
SD,<*hex value*>	Discovery profile mask.	FF
SF,1	Factory defaults.	N/A
SK,<*hex value*>	Connection profile mask.	FF
SN,<*string*>	Device name.	RN52-*XXXX*
SP,<*string*>	Pin code.	1234

Table A-2 describes the get (or display) commands.

TABLE A-2: GET (DISPLAY) COMMANDS

Command	Description
D	Basic settings.
G<*command*>	Displays setting for the set command indicated by <*command*>.
H	Display help.
V	Display the firmware version.

Table A-3 describes the action commands.

TABLE A-3: ACTION COMMANDS

Command	Description
+	Toggle the local echo of RX characters in command mode.
@,<*flag*>	Toggle whether the module is discoverable.
A,<*telephone number*>	Initial a voice call to <*telephone number*>.
B	Reconnect Bluetooth profiles to the mnost recently paired and connected device.
C	Accept an incoming voice call.
E	Terminate an active call or reject an incoming call.
F	Force a disconnect.
HV,<*value*>	The module sends a volume adjustment command to the telephone.
K,<*hex value*>	Kill the currently active connection.
M,<*flag*>	Toggle the on hold/mute function.
Q	Query the corrent connection status.
R,1	Reboot.
Y,<*flag*>	Return either the last speaker or microphone level.

Table A-4 describes the audio commands.

TABLE A-4: AUDIO COMMANDS

Command	Description
AV+	Increase the volume.
AV-	Decrease the volume.
AT+	PLay the next track.
AT-	Play the previous track.
AP	Pause or start playback.

Appendix B. Firmware Revision History

The following sections provide the firmware revision history.

B.1 VERSION 1.05

- First release.

NOTES:

Appendix C. Document Information

CONVENTIONS USED IN THIS GUIDE

This manual uses the following documentation conventions:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic characters	Referenced books	*MPLAB® IDE User's Guide*
	Emphasized text	...is the *only* compiler...
Initial caps	A window	the Output window
	A dialog	the Settings dialog
	A menu selection	select Enable Programmer
Quotes	A field name in a window or dialog	"Save project before build"
Underlined, italic text with right angle bracket	A menu path	*File>Save*
Bold characters	A dialog button	Click **OK**
	A tab	Click the **Power** tab
N'Rnnnn	A number in verilog format, where N is the total number of digits, R is the radix and n is a digit.	4'b0010, 2'hF1
Text in angle brackets < >	A key on the keyboard	Press <Enter>, <F1>
Courier New font:		
Plain Courier New	Sample source code	`#define START`
	Filenames	`autoexec.bat`
	File paths	`c:\mcc18\h`
	Keywords	`_asm, _endasm, static`
	Command-line options	`-Opa+, -Opa-`
	Bit values	`0, 1`
	Constants	`0xFF, 'A'`
Italic Courier New	A variable argument	*`file`*`.o`, where *`file`* can be any valid filename
Square brackets [ ]	Optional arguments	`mcc18 [options]` *`file`* `[options]`
Curly braces and pipe character: { | }	Choice of mutually exclusive arguments; an OR selection	`errorlevel {0|1}`
Ellipses...	Replaces repeated text	`var_name [, var_name...]`
	Represents code supplied by user	`void main (void) { ... }`

RECOMMENDED READING

This user's guide describes how to configure Roving Networks Bluetooth modules. The module-specific data sheets contain current information on the module specifications. Other useful documents are listed below. The following Microchip documents are available and recommended as supplemental reference resources:

RN52 Bluetooth Audio Module Data Sheet

This document provides the technical specifications for the RN52 module.

RN-52-EK Evaluation Kit User's Guide

This document describes how to use the RN-52-EK evaluation kit and provides an audio demonstration.

To obtain any of these documents, visit the Microchip web site at www.microchip.com.

DOCUMENT REVISION HISTORY

Version 1.0 (January 2013)

This is the initial released version of the document.