



Microchip Technology Inc.  
2355 West Chandler Blvd.  
Chandler, Arizona 85224-6199  
480-792-7200

# RN52 Bluetooth Audio Module

## Command Specifications

Revision History			
REV	DATE	Author	DESCRIPTION OF CHANGE
A	2/11/2013	Dale Loia	Initial Draft. Replaces “RN-52 commanset.95.docx”
B	2/14/2013	Dale Loia	Removed F command, table 1 corrections, state machine diagram
C	2/16/2013	Dale Loia	Fixed error in table
D	2/18/2013	Dale Loia	Re-ordered E command, “-“write only,
E	3/13/13	Dale Loia	New commands
F	3/26/13	Dale Loia	Changed I/O configuration commands from action to SET parameters.
G	3/26/13	Dale Loia	Clarified parameter size and format for I/O commands.
H	3/29/13	Dale Loia	Firmware Release 1.09: Restored I/O commands as actions, added more tables and examples
I	4/16/13	Dale Loia	Firmware Release 1.11: Updated release history; updated extended features “S%”
J	5/12/13	Dale Loia	Firmware Release 1.13.03 Hardened B and K,FF, and timeout flag the connection commands.



## Table of Contents

1	Introduction .....	4
2	GPIOs	4
2.1	Status Change Notification via PIO2 .....	4
2.2	Command Mode via PIO9.....	5
3	Event/Status Register .....	6
4	Set/Get Commands.....	7
5	Action Commands.....	9
6	Using GPIO .....	12
7	DFU – Device Firmware Update .....	12
7.1	Using DFU Loader utility .....	13
8	Factory Default Using PIO4.....	13

# 1 Introduction

This document describes the operation of the RN-52 Bluetooth Audio module command interface. The command interface operates over the Serial UART port. The UART has 2 modes, (1) DATA mode and (2) COMMAND mode.

In DATA mode, the UART operates as a raw serial bi-directional data pipe when connected to a remote BT client using SPP or Wireless iAP profiles.

## 2 GPIOs

The RN52 has 11 GPIO lines. Several GPIO are reserved for specific functions during boot-up and run time. The table 1 below lists the GPIOs and their functions. Two GPIOs are used at

Table 1: Default GPIO Assignment and Functions

GPIO	Function	Use
2	Output: Toggles from HIGH to LOW for 100ms to indicate a state change in the RN52. MCU should enter CMD mode and poll state register using "Q" action command. This IO is reserved and NOT available for general use at runtime.	Runtime, RESERVED
3	INPUT @ bootup: enter DFU MODE at boot time if USB device powers VBUS. PIO3 requires 47K to ground and 22K to USB VBUS if USB VBUS is supplying power to main board.	Bootup, Config
4	INPUT @ boot up: enter factory reset mode if driven LOW.	Bootup, Config
5	INPUT @ runtime, used for VOL-UP button. LOW is active	Runtime, Config
6	INPUT/OUTPUT at runtime	Runtime, Config
7	INPUT @ boot time, driven LOW will set UART baud rate to 9600, HIGH default baud 115200	Bootup, Config
9	INPUT: Driven LOW will put UART in CMD mode, other left to float HIGH, UART is in data mode. This IO is reserved and NOT available for general use at runtime.	Runtime, RESERVED
10	INPUT @ runtime, used for VOL-DN.button, LOW is active	Runtime, Config
11	INPUT @ runtime, used for PREV-TRACK button, LOW is active	Runtime, Config
12	INPUT @ runtime, used for NEXT-TRACK button, LOW is active	Runtime, Config
13	INPUT @ runtime, used for PLAY/PAUSE button, LOW is active	Runtime, Config

### 2.1 Status Change Notification via PIO2

The RN52 drives PIO2 (as an output) to notify an external MCU of an event or status change. Status changes include changes to profile connections or voice call connections. The MCU

should enter command mode and issue the “Q” command to retrieve latest status. See Section 3 for more information regarding status changes.

### Firmware 1.02 to 1.11

PIO2 is held LOW for 100ms to indicate a change, or event, has occurred. The timing diagram below illustrates three event notifications.

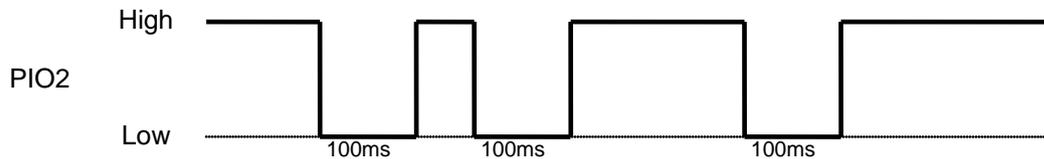


Figure 1: PIO2 Event Notification 1.12 and below

### Firmware 1.12 or later

Starting with firmware 1.13 and later, PIO2 is held LOW for each new event and remains latched LOW until “Q” command unlatches PIO2 back to HIGH.

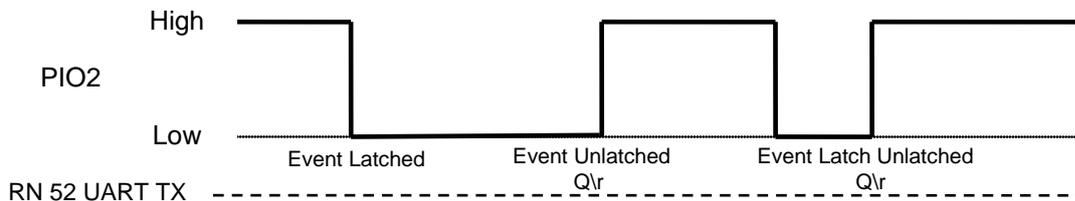


Figure 2: PIO2 Event Notification

## 2.2 Command Mode via PIO9

The RN52 monitors PIO9 (as an input) to determine whether data traversing the UART is routed to active Bluetooth SPP (serial port profile) connection or to the command console. When PIO9 is held high, the UART enters command mode. In command mode all data enters UART is routed to command console where the module can be configured and queried for status using the ASCII commands.

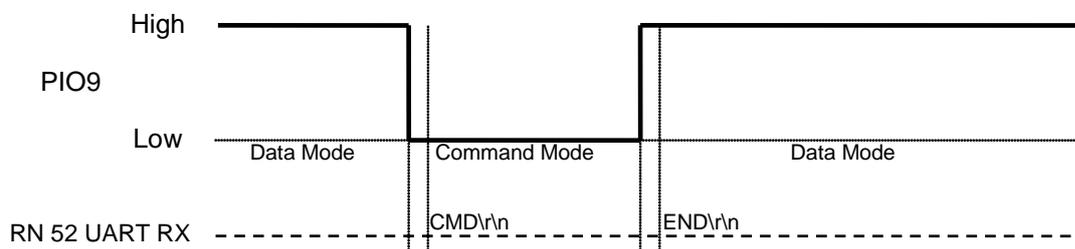


Figure 3: PIO9 pull low routes UART traffic to command console

For applications when an MCU controls PIO9 and commands over UART, the MCU should monitor UART RX line for the following five character strings: “CMD\r\n” and “END\r\n” immediately after driving PIO9.

Use the following state diagram as a guideline for designing code to monitor PIO9 and command states.

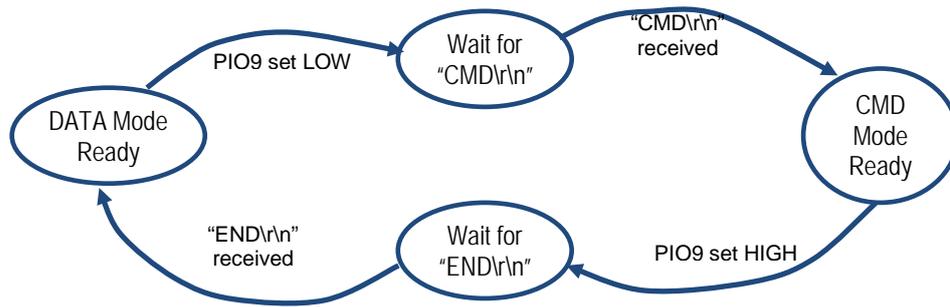


Figure43: State Diagram for entering and exiting CMD mode

### 3 Event/Status Register

The “Q\r” action command returns a multi-byte list, where each value is expressed as “00”-“FF”. The byte list is ordered little endian (byte 0 LSB, byte N last) terminated by carriage return and line feed characters (\r\n).

Table 1 describes the current status bytes. For example, the byte string “0C16\r\n” indicates A2DP and HFP profiles are connected and connection state is “ActiveCall (6)” and Audio Mute/Hold event was received from phone.

**Connection state value (bits 0-3) do NOT change unless the module status changes.**

**Event bits 4-6 in BYTE[1] are cleared when “Q” is issued.**

**The module drives GPIO2 LOW for 100ms to notify Docking Station that the event/status register has been changed.**

Table 1: Profile Status and Connection

BYTE0	
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
BYTE1	
Bit	Description
0-3	Connection States: 0-15, see table 2 below
4	HFP audio volume level change from audio gateway (phone), use “Y,0” to retrieve volume level
5	HFP audio microphone level change from audio gateway (phone), use “Y,1” to retrieve microphone level
6	HFP Voice mute/hold event triggered by Phone
7	Connection timeout flag: used to indicate if “B” re-connect command time out. This flag is cleared

	after “Q” status is returned.

Table 2: Connection States in Byte 1, bits 0-3

Value	State	Description
0	Limbo	logically off but physically on
1	Connectable	connectable - page scanning
2	Connectable and Discoverable	connectable and discoverable - page and inquiry scanning
3	Connected	connected to an AG
4	Outgoing Call Establish	connected AG has an outgoing call in progress
5	Incoming Call Establish	connected AG has an active call in progress and the audio is in the headset
6	Active Call On Handsfree	connected AG has an active call in progress and the audio is in the headset
7	Test Mode	headset is in test mode
8	Three Way Call Waiting	connected AG has an active call and a second call on hold
9	Three Way Call On Hold	connected AG has an active call and a second call on hold
10	Three Way Call Multi Call	connected AG has an active call and a second call on hold
11	Incoming Call On Hold	connected AG has an incoming call on hold
12	Active Call On Handset	connected AG has an active call and the audio is in the handset
13	Audio Streaming	headset is streaming A2DP audio
14	Low Battery	low battery state

## 4 Set/Get Commands

The following parameters can be written and read from command console using SET & Get commands. SET and GET use the following syntax where the parameter is a single character.  
:

“S<char>,<value>\r”	Set a parameter
“G<char>\r”	Read a parameter

The parameter <char> is a single character. All SET commands are non-blocking and will return immediately. **Set commands DO NOT TAKE EFFECT until module is rebooted**

with “R,1” command unless noted otherwise. Not all set parameters are readable with a get.

All commands respond with “AOK, <opt msg>\r\n” for success, or “ERR, <opt msg>\r\n” if command fails. <opt msg> is optional message text.

Table 3: Set/Get parameters

Parameter	Description
A, <decimal>	<b><u>Authentication Mode</u></b> Example: “SA,4\r” set module to legacy PIN code. <decimal> 0=open, 1=keyboard I/O, 2=just works, 4=pin code, Default: 0x1 (keyboard)
C, <hex>	<b><u>Class of Device</u></b> Set COD for Bluetooth device. This is 24-bit (6 byte) hex value. Default is 240407, corresponding to
D, <byte>	<b><u>Discovery Mask</u></b> Byte value to enable specific profiles that can be discovered by remote Bluetooth hosts. See Byte0 definition for profile value. Default is FF, all profiles.
F, 1	<b><u>Factory Default</u></b> Sets all parameters to factory default values. The changes do not take effect until module is rebooted. This is a write-only parameter. Example: “SF,1\r” invokes factory default.
K, <byte>	<b><u>Connection Mask</u></b> Byte value to enable specific profiles that can be connected by remote Bluetooth hosts. See Byte0 definition for profile value. Default is FF, all profiles.
N, <string>	<b><u>Bluetooth Name</u></b> Sets the Bluetooth friendly name advertised during an inquiry scan. Name can be up to 16 characters.
P, <string>	<b><u>PIN Code</u></b> Sets four digit PIN code used for legacy pairing.
-, <string>	<b><u>Serialized Name</u></b> Example: “S-,AUDIO\r” sets Bluetooth name to friendly address “AUDIO-b1b0” where b0 and b1 hex values for first two bytes (right most) of MAC address i.e. “AUDIO-51A2”. NOTE: this is a write only parameter
, <byte>	<b><u>Audio Route</u></b> Set the route for audio output. Values are: 00 - Analog (default)

	01 - I2S (variable sample rate, 24bit sample, left justified, 1 delay) 02 - S/PDIF Example: "S ,02\r" set audio output to SPDI/F at next reboot.
^,<dec>	<b>Automatic Shutdown on Idle</b> 0- Feature disable, on forever 1-600 seconds of idle time (not-connected)
%,<hex8>	<b>Extended Feature</b> Bit 0 - enable AVRCP buttons for EK Bit 1 - enable reconnect on power-on Bit 2 - discoverable on start up NOTE: Extended features default value is set at factory to 00 for RN52 modules and 07 for RN52EK. The "SF,1" does NOT change this value. It must be changed with "S%,<hex8>" command.

## 5 Action Commands

Action commands make immediate changes to the module and do NOT require reboot. The format of action command is as follows:

<char>,<optional argument>\r

All Action commands are non-blocking and will return immediately. If background processing is required then event notification should be used to determine if operation succeeded or failed.

All commands respond with "AOK,<opt msg>" for success, or "ERR,<opt msg>" if command fails. <opt msg> is optional message text.

	Description
A,<phone>	<b>Initiate Voice Call</b> Sends command to Phone to initiate a call to <phone>, where phone is a string max 25 characters; example "A,14083955300", ERR returned if connection state is NOT (0,1,2).
AV+	<b>Volume UP</b> Sends AVRCP volume up command to AG or BT host.
AV-	<b>Volume UP</b> Sends AVRCP volume down command to AG or BT host.
AT+	<b>Next Track</b> Sends AVRCP next track command to AG or BT host.
AT-	<b>Previous Track</b> Sends AVRCP previous track command to AG or

	BT host.
AP	<b><u>Play/Pause Toggle</u></b> Sends AVRCP play/pause command to AG or BT host.
B	<b><u>Reconnect to Profiles</u></b> Attempt to reconnect profiles to most recently paired & connected device. Reconnect will be attempted on the profiles specified in the connection mask. See the “SK,<value>” command. Use “Q” to retrieve BT profile connection status in byte 0 (bits 0-3) value. ERR returned if module has not been previously connected or connection mask is set to 00 (00 means module is unconnectable).
C	<b><u>Accept Incoming Voice Call</u></b> Use “Q” to retrieve call status Byte[1]:bits 0-3 value. ERR returned if call status is NOT set to incoming call established (5).
D	<b><u>Display Active Configuration</u></b> Display configuration settings.
E	<b><u>Drop or Reject Call</u></b> Terminate an active call or reject an incoming call. Call status must be either (1) Incoming call, (2) active call, otherwise ERR is returned.
H	<b><u>Display Help Text</u></b> Displays help text. Useful when accessing command console from terminal emulator.
HV,<decimal>	<b><u>Set AG Volume Level</u></b> Sends volume level to AG (audio gateway) to adjust voice call volume. This command is used to synchronize volume levels. Level in decimal integer 0-15; example “HV,8”. Conform to HFP Specification 1.6 4.28.2.
I@ I@,<hex16>	<b><u>Configure GPIO direction</u></b> 16-bit mask used to set GPIO direction, bit on for output, bit off for input If no argument specified the current configuration is returned
I& I&,<hex16>	<b><u>GPIO level and input pull-up configuration</u></b> Bit on is logical high, bit off logical low If no argument specified the current levels are returned. The 16-bit <hex16> value is PIO bit map (bit4 = PIO4). When PIO is configured as output, I&,<hex16> drives the level high for PIO specified in

	<p>&lt;hex16&gt;. When PIO is configured as input, I&amp; configures the PIO to use a pull-up (bit set) or pull-down(bit clear).</p>
K,<byte>	<p><b><u>Disconnect Profiles</u></b> Drop connections specified on profile byte (see Byte 0, bit 0-3)</p>
Q	<p><b><u>Show Connection Status</u></b> Display byte (ascii hex) string indicating connected profiles, connection states, and other events. See section 3.</p>
M,<enable>	<p><b><u>Hold/Mute Current Call</u></b> Places active call on hold/mute or toggles hold/mute. “M” toggles hold/mute on/off “M,0” - unmute call. ERR returned if call status is NOT Active Call (2).</p>
R,1	<p><b><u>Reboot Module</u></b> Drops all active connections and reload stored values from previous “S” (set) commands.</p>
Y,<0,1>	<p><b><u>Gain Level (Speaker or Microphone)</u></b> In response to bit 14 &amp; 15 in event status register, this command returns either last speaker or microphone level as 0-15 decimal value sent from audio gateway. “Y,0” returns speaker volume “Y,1” returns microphone level</p>
V	<p><b><u>Display Firmware Version</u></b> Displays firmware level.</p>
@,<decimal>	<p><b><u>Discoverability Enable/Disable</u></b> Enables or disables discoverability of module. “@,0”, disable discoverability “@,1”, enable discoverability</p>
+	<p><b><u>Local Echo Toggle</u></b> Helpful when accessing the command console from terminal emulator</p>

## 6 Using GPIOs

The RN52 GPIO are configurable as action commands. The IO configuration does not persist and must be through “I@” and “I&” commands when module is running.

**A 16-bit mask is used to refer each GPIO in the “I@” an “I&” commands. The default state is undefined until they are configured. Default Mask as boot time. The available GPIO are shown below. Any GPIO grayed out row is not available for general purpose use.**

### GPIO MAP

PIO	Mask	Description
0	0001	N/A
1	0002	N/A
2	0004	RESERVED, output only used to drive state change notifications. The mask value “I@” will always return “0004”
3	0008	
4	0010	
5	0020	Volume up in AVRCP mode
6	0040	
7	0080	
8	0100	N/A
9	0200	RESERVED, input only used to detect CMD mode entry.
10	0400	Volume down in AVRCP mode
11	0800	Previous track in AVRCP mode
12	1000	Next Track in AVRCP mode
13	2000	Pause/Play in AVRCP mode
14	4000	N/A
15	8000	N/A

### **WARNING for RN52-EK (AVRCP mode enabled):**

It is not recommended to enable the “S%,01” AVRCP buttons and configure the button PIOs (5,10-13) as outputs (i.e. “I@,3C20”). Pressing the buttons in this configuration may damage the RN52 module.

### **PIO INPUT/OUTPUT EXAMPLE:**

This example assume the AVRCP mode is disabled (S%,00). To configure GPIO 5-7 as inputs with pull up, and 12,13 outputs, issue the following commands.

```
“I@,3000\r” // configured 12 & 13 as outputs, the rest are inputs
“I@\r” // read the PIO mask
“3004\r\n” // PIO2 (bit2=0004) will always be configured as output
“I&,20E0\r” // enable internal pull up resistor on 5,6,7,drive 13 high, clear
remaining levels
```

## 7 DFU – Device Firmware Update

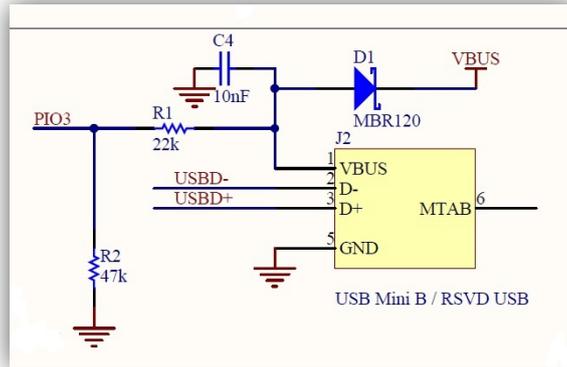


Figure 5: schematic USB DFU port and PIO3

Device Firmware Update (DFU) uses the USB port to update the firmware in the RN52. Implement the DFU feature is highly encouraged as firmware updates offer new features which enhance the functions of the RN52 module. Follow the reference design in the figure X. When an external USB host is plugged into DFU port and energized VBUS, the RN52 enters DFU mode when module is rebooted or powered up. The line C4 and D1 to VBUS input on voltage regulator (see full reference design schematic in data sheet) is not need if customer board is not being power from USB VBUS.

**A 47K pull down resistor (See R2 on figure x) is required on PIO3 even if DFU USB reference is not used.**

### 7.1 Using DFU Loader utility

TBD

## 8 Factory Default Using PIO4

[step by step] TBD

## 9 Firmware Release Revision

Version	Description	
1.08	First release on RN52-EK, Mar 2013	Mar 2013
1.09	Added new commands: “I@” action command to configure GPIO “I&” action command to set/get GPIO levels “S%”, to configure extended features	Apr 2013
1.10	Modified “S%” extended features to avoid factory default via “SF,1” command. Extended features must be set with “S%,<byte>” command. This was done to support RN52EK configuration on production line.	Apr 2013
1.11	Fixed a issue with “B” command not reconnecting iAP profile	Apr 2013
1.12	Changed GPIO2 from pulse to latched/unlatch mechanism	Apr 2013
1.13	Hardened K,FF and B, a	May 2013