RN4678 Bluetooth® Dual Mode Module Command Reference User's Guide

© 2016-2019 Microchip Technology Inc.

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949 ═

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, AVR, AVR logo, AVR Freaks, BitCloud, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, Kleer, LANCheck, LINK MD, maXStylus, maXTouch, MediaLB, megaAVR, MOST, MOST logo, MPLAB, OptoLyzer, PIC, picoPower, PICSTART, PIC32 logo, Prochip Designer, QTouch, SAM-BA, SpyNIC, SST, SST Logo, SuperFlash, tinyAVR, UNI/O, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, EtherSynch, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and Quiet-Wire are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Adjacent Key Suppression, AKS, Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, CryptoAuthentication, CryptoAutomotive, CryptoCompanion, CryptoController, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, INICnet, Inter-Chip Connectivity, JitterBlocker, KleerNet, KleerNet logo, memBrain, Mindi, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PowerSmart, PureSilicon, QMatrix, REAL ICE, Ripple Blocker, SAM-ICE, Serial Quad I/O, SMART-I.S., SQI, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GestIC is a registered trademarks of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

©2016-2018, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-5224-4312-4

© 2016-2019 Microchip Technology Inc.

Table of Contents

Preface **5**

Chapter 1. Overview

1.1 Introduction 9
1.2 Command Mode vs. Data Mode 9
1.3 Control RN4678 over UART 9
1.4 Operating Modes 10
1.5 Security Modes 10

Chapter 2. Command Reference

2.1 Introduction 11
2.2 Command Syntax 11
2.3 Set Commands 11
2.4 Get Commands 21
2.5 Action Commands 22

Appendix A. Command Quick Reference Guide

A.1 Default Configuration Settings 31

Worldwide Sales and Service **33**

© 2016-2019 Microchip Technology Inc.

NOTE:

© 2016-2019 Microchip Technology Inc.

Preface

NOTICE TO CUSTOMERS

All documentation becomes dated, and this manual is no exception. Microchip tools and documentation are constantly evolving to meet customer needs, so some actual dialogs and/or tool descriptions may differ from those in this document. Please refer to our web site (www.microchip.com) to obtain the latest documentation available.

Documents are identified with a "DS" number. This number is located on the bottom of each page, in front of the page number. The numbering convention for the DS number is "DSXXXXXXXXA", where "XXXXXXXX" is the document number and "A" is the revision level of the document.

For the most up-to-date information on development tools, see the MPLAB® IDE online help. Select the Help menu, and then Topics to open a list of available online help files.

INTRODUCTION

This chapter contains general information that will be useful to know before using the RN4678 Bluetooth® Dual Mode Module. Items discussed in this chapter include:

- Document Layout
- Conventions Used in this Guide
- Recommended Reading
- The Microchip Web Site
- Development Systems Customer Change Notification Service
- Customer Support
- Document Revision History

DOCUMENT LAYOUT

This document provides information for configuring the RN4678 module, including a command reference, advanced features, and application examples. The document is organized as follows:

- **Chapter 1. "Overview"** - This chapter describes the operating modes of the RN4678 Bluetooth® Dual-Mode Module.
- **Chapter 2. "Command Reference"** - This chapter describes the general command categories of the RN4678 Bluetooth® Dual-Mode Module in detail.
- **Appendix A. "Command Quick Reference Guide"** - This Appendix provides a quick and summarized overview of the commands discussed in the previous chapter.

© 2016-2019 Microchip Technology Inc.

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 brackets 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) { ... }`

© 2016-2019 Microchip Technology Inc.

RECOMMENDED READING

This user's guide describes how to use the RN4678 Bluetooth® Dual-Mode Module. Other useful document(s) are listed below. The following Microchip document(s) are recommended as supplemental reference resources.

RN4678 Bluetooth® Dual Mode Module Data Sheet (DS50002519)

This document provides the technical specifications for the RN4678 module and is available for download from the Microchip website (www.microchip.com)

THE MICROCHIP WEB SITE

Microchip provides online support via our web site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events; and listings of Microchip sales offices, distributors and factory representatives

DEVELOPMENT SYSTEMS CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer The Development Systems product group categories are:

- **Compilers** – The latest information on Microchip C compilers and other language tools
- **Emulators** – The latest information on the Microchip MPLAB® REAL ICE™ in-circuit emulator
- **In-Circuit Debuggers** – The latest information on the Microchip in-circuit debugger, MPLAB ICD 3
- **MPLAB X IDE** – The latest information on Microchip MPLAB X IDE, the Windows® Integrated Development Environment for development systems tools
- **Programmers** – The latest information on Microchip programmers including the PICkit™ 3 development programmer

© 2016-2019 Microchip Technology Inc.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at:

http://www.microchip.com/support.

DOCUMENT REVISION HISTORY

Revision A (June 2016)

This is the initial release of this document.

Revision B (October 2016)

This revision includes the following updates:

- Modified **Section 1.3 "Control RN4678 over UART"** to update port settings to Hardware flow control enabled.
- Modified Section 2.3.19 to add a sentence to the existing 0x1000 description in Table 2-5, and to change its default text value to 1000.

Revision C (December 2018)

This revision includes the following updates:

- Modified **Section 2.3.17**, "SO" command to suppress the status strings.
- Modified **Section 2.3.19** to describe the "SQ, 0008" command which can enable the SW_BTN.
- Added **Section 2.3.23** and **Section 2.3.24** for new commands STO and STD which are used to configure the power on standby timeout and disconnect standby timeout respectively.
- Modified **Section 2.3.25** to update the baud rate settings, added support for 1Mbps baud rate.
- Modified **Section 2.4.1**, "D" command to display the feature value which is set by "SQ" command
- Modified **Section 2.5.11** to add description of for the F command response format.
- Added **Section 2.5.18** for the new command O,0 which puts the module in dormant mode.

© 2016-2019 Microchip Technology Inc.

Chapter 1. Overview

1.1 INTRODUCTION

Microchip RN4678 Certified Bluetooth Dual Mode RF module supports both Bluetooth Classic and Bluetooth Low Energy (BLE) communication. Under Bluetooth Classic, RN4678 implements the standard Serial Port Profile (SPP) that supports data streaming between two Bluetooth Classic devices. Under BLE, RN4678 introduces a custom private service that mimics the functionality of SPP and supports data streaming between two BLE devices. This BLE data streaming function in RN4678 is labeled as UART Transparent. The data streaming functionality of RN4678 under Bluetooth Classic and BLE behaves virtually the same, creating a universal user experience across two different communication protocols.

The RN4678 module is developed on top of the BM78 module. Its hardware is identical to BM78 and pin-compatible with BM77 and RN4677. A major difference between the RN4678 and BM78 is that RN4678 provides the control interface based on ASCII commands. The command interface on RN4678 is similar to that of previous RN modules, such as RN41/RN42, RN52, and RN4020, providing an easy migration path for customers who are currently using RN Bluetooth modules. The command interface on the RN4678 is also very similar to that of the RN4677, with minimum syntax modifications. Along with BM78, RN4678 additionally provides support for the iAP2 protocol over Bluetooth classic to provide data streaming capability with latest Apple iOS® devices. Apple requires any new product developed under Made for iOS (MFi) product to follow the iAP2 protocols.

1.2 COMMAND MODE VS. DATA MODE

The RN4678 module operates in two modes: Data mode (default) and Command mode. When RN4678 is connected to another device and in Data mode, RN4678 acts as a data pipe: anything received from UART is passed to the connected peer device through SPP if connected to a Bluetooth Classic device, or via a custom private service, UART Transparent, if connected to a BLE device. When data is received from the peer device via SPP for Bluetooth Classic or UART Transparent for BLE, such data outputs directly to UART.

The RN4678 module is configured or controlled, or both, by setting it into Command mode and executing ASCII commands over UART. All configuration changes made by Set commands remain in Non-Volatile Memory (NVM) and survive the power cycle. Any configuration changes made need a reboot of module to take effect. All Action commands take effect immediately, but have *no* effect in case of power recycle.

1.3 CONTROL RN4678 OVER UART

A terminal emulator, such as TeraTerm (Windows) or CoolTerm (Mac OS-X®), can be used to control the RN4678 PICtail™ board from a computer.

With the RN4678 PICtail board connected to a computer and a serial port enumerated for the UART port, run the terminal emulator to open the COM port with the following port settings:

- Baud rate 115,200 bps

- 8 bits
- No Parity
- 1 Stop bit
- Hardware flow control enabled

To enter Command mode, type `$$$` into the terminal emulator. Once RN4678 enters Command mode, it sends the string `CMD` via the UART to indicate the start of Command mode session. Once in Command mode, valid ASCII commands can be issued to control or configure RN4678. All commands need to be terminated with a carriage return('\r') character. The RN4678 provides a response to every command entered. The next command must *not* be issued before having received a response for the previous command. For Set or Action commands, the positive response is `AOK` and negative response is `ERR` followed by an error code. All error codes are listed in Table 1-1. By default, when the RN4678 is ready to receive the next command, the command prompt `CMD>` is sent via the UART. Users can turn off this feature by using the command `SQ` to ensure that the interface is backward compatible with RN41/42/52/4020.

TABLE 1-1: LIST OF ERROR CODES

Error Code	Error Message	Description
1	ERR1	Error in command syntax
2	ERR2	Error in Parameter format or range
3	ERR3	Error in accepting command in current operating state
4	ERR4	Error in current settings

In Command mode, if the <ESC> key (ASCII code 0x1B) is sent to UART, all previous and unexecuted inputs are ignored, and a new command prompt displays when the module is ready to take a new command. To return to Data mode, type `---<cr>`.

1.4 OPERATING MODES

The RN4678 module has two operating modes that can be set by using the command `SM`.

- **Default mode** (`SM,0`) – In this mode, other Bluetooth devices can discover and connect to the module. Outbound connections can also be initiated in this mode.
- **Paring mode** (`SM,6`) – In this mode, the module attempts to connect to the remote device whose MAC address matches the value stored in the "Remote Address" field in the module. The command `SR` sets the "Remote Address" field.

1.5 SECURITY MODES

The RN4678 module supports encryption and authentication with security modes from 1 to 4. The definitions of Security modes are as follows:

- Security mode 1: Passkey Confirm
- Security mode 2: Just Works
- Security mode 3: Passkey Entry
- Security mode 4: Legacy Pin Code

For Bluetooth Classic, all of these security modes are supported. For BLE, *only* security modes from 1 to 3 are supported.

For Security mode 1 and Security mode 3, by default, a random 6-digit security pin is generated and displayed at one end of the connection and the other end must enter the security pin. Optionally, in Security mode 1, it is possible to fix the 6-digit security pin for BLE connection by supplying a 6-digit pin using the command `SP` instead of a 4-digit pin for Legacy Pin Code mode.

© 2016-2019 Microchip Technology Inc.

Chapter 2. Command Reference

2.1 INTRODUCTION

The RN4678 module supports a variety of commands for controlling and configuration. This section describes these commands in detail and provides examples.

2.2 COMMAND SYNTAX

The ASCII command syntax is a keyword followed by optional parameters.

- ASCII commands are divided into three groups:
  - Set Commands
  - Get Commands
  - Action Commands
- All commands contain one, two or three case-insensitive characters
- Delimit command and parameter(s) with a comma
- Text data is case sensitive, such as Bluetooth name
- All commands end with carriage return (`'\r'`, `\x0d`)
- Get commands return the value to be retrieved. Most of the other commands return either `AOK` as a positive response or `ERR` as a negative response.

2.3 SET COMMANDS

Set commands change the RN4678 configurations and take effect after rebooting via `R,1` command, hard Reset or power cycle.

2.3.1 SA,<1-4>

The Set Authentication command sets the authentication method when a remote device attempts to connect, where *<1-4>* is one of the values shown in Table 2-1.

Once a remote device exchanges pin codes with the RN4678 module, a link key is stored for future use. The device automatically and permanently stores up to eight peer devices in Flash memory, using the First-In, First-Out (FIFO) method.

TABLE 2-1: SET AUTHENTICATION VALUES

Value	Description
1	Secure Simple Pairing (SSP) Pin Code Confirm mode If this option is set, a 6-digit pin is sent to UART when authenticating the link. On the remote side, the same 6-digit pin is displayed and confirmation is requested. For BLE, if the command `SP` sets the 6-digit pin code, the pin code is displayed and confirmed on the remote side; otherwise, a random 6-digit pin code is generated for confirmation.
2	*(Default)* SSP "Just Works" mode This mode works without any request to display or input any security pin.

© 2016-2019 Microchip Technology Inc.

TABLE 2-1: SET AUTHENTICATION VALUES (CONTINUED)

Value	Description
3	SSP Pin Code Input mode If this mode is set, a 6-digit pin is displayed at the remote device when authenticating the link and the local RN4678 device is required to input the pin.
4	Legacy Pin Code mode This Security mode is defined in *Bluetooth Specification 2.0* and in the process of being phased out. *Only* Bluetooth Classic supports this Security mode. This mode requires a fixed 4-digit pin code that can be configured by using the command `SP`.

Default: 2

Example: `SA,2` // Enable "Just Works" security mode

2.3.2 SC,<H16>

This command *only* applies to Bluetooth Classic.

This command sets the "Service Class" field in the Class of Device (COD). The service class consists of the 11 Most Significant bits in the COD. This command sets the most significant word (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 at www.bluetooth.org.

Default: 0000

Example: `SC,0002` // Set service class to 0002

2.3.3 SD,<H16>

This command *only* applies to Bluetooth Classic.

This command sets the last significant word (lsw) of the Class of Device (COD). The COD is a 24-bit number. The higher 8-bit value is set by using the command `SC` and the lower 16-bit value is set by using the command `SD`.

Default: 1F00

Example: `SD,8040`

To set the COD to 0x1F0123, use the following commands:

```
SC,001F
SD,0123
```

2.3.4 SDM,<TXT>

This command *only* applies to BLE.

This command sets the model string in the BLE Device Information service. It expects the model number string which can be up to 16 characters as the input parameter

Default: RN4678

Example: `SDM,MyModel`

2.3.5 SDN,<TXT>

This command *only* applies to BLE.

© 2016-2019 Microchip Technology Inc.

This command sets the manufacturer string in BLE Device Information service. It expects the manufacturer string which can be up to 16 characters as the input parameter.

Default: Microchip

Example: `SDN,MyManufacturer`

2.3.6 SDR,<TXT>

This command *only* applies to BLE.

This command sets the software revision of the firmware. It expects an ASCII string which can be up to four characters as software revision.

Default: Current RN4678 software version

Example: `SDR,1.23`

2.3.7 SDS,<TXT>

This command *only* applies to BLE.

This command sets the serial number of the device. It expects an ASCII string which can be up to 16 characters as the serial number of the device.

Default: MAC address of the device

Example: `SDS,1234567890`

2.3.8 SE,<H16>

This command *only* applies to Bluetooth Classic.

This command sets the Universally Unique Identifier (UUID) for service discovery process. It *only* supports standard 16-bit UUID.

Default: 1101

Example: `SE,1101` // Set UUID for serial port

2.3.9 SF,1

This command restores the device to the factory defaults.

Default: N/A

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

2.3.10 SG,<0-2>

This command changes Bluetooth modes. It accepts any one of the parameter values listed in Table 2-2 to set the Bluetooth mode.

TABLE 2-2: BLUETOOTH MODES

Parameter Value	Bluetooth Mode
0	Dual mode
1	Bluetooth Low Energy *only*
2	Bluetooth Classic *only*

Default: 0

Example: `SG,1` // Set RN4678 to be BLE *only*

2.3.11 SH,<0,1>

This command enables low-power operation of the RN4678. It expects a single digit input parameter. If the input parameter is 1, the RN4678 uses a 16 MHz clock in Idle mode and UART is always operational. If the input parameter is 0, the RN4678 uses a 32 kHz clock in Idle mode and thus has lower power consumption, but the UART_RX_IND pin must be pulled low to switch to the 16 MHz clock before using UART.

Default: 1

Example: `SH,0` // Enable low-power operation

2.3.12 SI,<H16>

This command *only* applies to Bluetooth Classic.

The Inquiry Scan Window command sets the length of time the device spends enabling an inquiry scan (discoverability). The minimum value is `0x0012`, corresponding to about a 1% duty cycle. The page scan interval is fixed at `0x1000`. The default window is `0x0100`. If the host is already paired, the inquiry scan is *not* used.

Default: 0100

Example: `SI,0200` // Set inquiry scan window to 0x0200

2.3.13 SJ,<H16>

This command *only* applies to Bluetooth Classic.

The Page Scan Window command sets the amount of time the device spends enabling page scanning (connectable). The minimum value is `0x0012`, which corresponds to about a 1% duty cycle. The page scan interval is fixed at `0x1000`. The default window is `0x0100`. The maximum value is `0x800`. Set this option to `0x0000` to disable page scanning and render the device unconnectable.

Default: 0100

Example: `SJ,0200` // Set the page scan window to 0x0200

2.3.14 SL,<H8>

This command sets the duration of the Bluetooth Classic and BLE scan operation. It expects one 8-bit input parameter in hex format. The input parameter indicates the maximum scan duration in the unit of 10 seconds. The scan duration must be between `01` and `06`. Any scan duration beyond 60 seconds (`06` as input parameter) is considered to last forever. The scan operation can be terminated earlier if *no* memory is available to store scan results. Command `X` can be used to terminate the scan before the scan duration expires.

Default: 02 // Default scan duration is 20 seconds

Example: `SL,1` // Set scan duration to be 10 seconds

2.3.15 SM,<value>

This command sets the mode, where <*value*> is a number as shown in Table 2-3.

TABLE 2-3: MODE VALUES

Value	Description
0	Default mode
6	Auto Reconnect mode

© 2016-2019 Microchip Technology Inc.

Default: 0

Example: `SM,0` // Set the mode to Default mode

2.3.16 SN,<string>

This command sets the device name, where <*string*> is up to 16 alphanumeric characters. By default the device name (Classic/BLE) is set to RN4678-b1b0 where b1b0 are lower order byte values of device BT address.

Default: N/A

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

2.3.17 SO,<string1>[,<string2>]

This command sets the extended status string, where <*string1*> and <*string2*> can be up to four alphanumeric characters each. Setting these delimiter strings for the status strings from the UART can help parse the results. The first parameter, <*string1*>, is the prefix to the message and the optional second parameter, <*string2*>, is the postfix to the message. For instance, when a Bluetooth Classic connection is established, the device sends the string `<string1>CONNECT<string2>`.

If there are no parameters passed as `<string1> and <string2>` (SO,,) then all the status strings in Table 2-4 are suppressed including the “Rebooting” print string response for `R,1` command. If the status strings are suppressed, the G0 command will return NULL string to indicate that there are no `<string1>` and `<string2>`.

TABLE 2-4: LIST OF STATUS STRINGS

Status String	Description
AUTHENTICATED	BT Classic iAP authentication success
AUTH_FAIL	BT Classic iAP authentication failure
BONDED	New bonding information saved
CONNECT	Device connected with peer
DISCONN	Device disconnect from peer
END_INQ	End of BT classic inquiry
END_SCN	End of BLE scan
ERR_CONN	Connection attempt fails
ERR_CONN_PARAM	BLE fails to update connection parameters
ERR_LSEC	BLE security failure
ERR_SEC	BT Classic security failure
FACTORY_RESET	Factory reset is triggered by factory reset pin in the first 5 seconds after powering up
LBONDED	BLE saved linked key
LSECURED	BLE secured link
LSECURE_FAIL	BLE security failure
LSTREAM_OPEN	BLE UART Transparent stream open
MLDP_MODE	RN4020 MLDP service detected
REBOOT	RN4678 reboot
RFCOMM_CLOSE	SPP stream service deactivated
RFCOMM_OPEN	SPP stream service activated
SECURE_FAIL	BT Classic security failure
SESSION_CLOSE	BT Classic iAP data session close
SESSION_OPEN	BT Classic iAP data session open

TABLE 2-4: LIST OF STATUS STRINGS

Status String	Description
SECURED	BT Classic secured link
IAP1_DISABLED	If a peer device tries to connect via iAP1 when RN4678 module has iAP1 feature disabled

Default: %,%

Example: `SO,<,>` // Set the prefix and postfix of the message to // "<" and ">", respectively

2.3.18 SP,<string>

This command sets the security pin code, where <*string*> is either a 4-digit or a 6-digit pin. The 4-digit pin is used for Legacy Pin Code pairing, while the 6-digit pin is used for SSP authentication in BLE if fixed pin is desirable. The fixed 6-digit pin code in BLE is *not* supported by many of the Bluetooth Low Energy implementations. The user must understand the security impact of using the fixed pin code under SSP mode.

Default: 1234

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

2.3.19 SQ,<H16>

This command sets the features of the RN4678 module. It expects one input parameter in the format of bitmap. The bitmap must be interpolated according to Table 2-5.

TABLE 2-5: FEATURE SET BITMAP

Bit Map	Description
0x8000	If set, RN4678 is set into Fast mode. Once set into Fast mode, the command `$$$` is *no* longer available to put the device into Command mode. The CMD/DATA switch pin defined in command `SX` controls the device operation mode.
0x4000	If set, *no* prompt `CMD>` is required when RN4678 is ready to accept a new command.
0x2000	If set, *no* acknowledgment is required for UART Transparent under BLE connection.
0x1000	If set, the hardware flow control for UART is enabled. Flow control is enabled by default.
0x0800	If set, *no* non-connectable beacon shows up in the BLE scan result.
0x0400	If set, *no* connectable advertisement shows up in the BLE scan result.
0x0200	If set, RN4678 does *not* filter out duplicate BLE scan results. Set this bit if the RN4678 expects to see dynamically changing advertisements.
0x0100	If set, RN4678 performs passive scan instead of default active scan.
0x0080	If set, RN4678 reboots after disconnection.
0x0008	If set, SW_BTN functionality is enabled. By default the SW_BTN is disabled.

Default: 1000 // Hardware flow control enabled by default

Example: `SQ,8000` // Enable Fast mode

`SQ,9080` // Enter Fast mode, flow control enabled, // reboots after disconnection

© 2016-2019 Microchip Technology Inc.

2.3.20 SR[,<0,1,Z>]<MAC Address>

This command stores the remote address. When the MAC address is for a Bluetooth Classic device, then the *only* parameter to be entered is the Bluetooth MAC address, a 12 hexadecimal digits (6 bytes) address without space or characters between digits. For a BLE device, the first parameter is either `0` (public) or `1` (random) to indicate address type, followed by the MAC address. Additionally, this command takes special character `Z` to erase any stored address. For BLE random address, link back feature may *not* work since the BLE address may have changed when performing the link back feature.

Default:	N/A	
Example:	`SR,00A053112233`	// Set the remote Bluetooth Classic // address to 00A053112233
	`SR,0,00A053112233`	// Set the remote BLE address to // 00A053112233
	`SR,Z`	// Clear stored address

2.3.21 SS,<string>

This command sets the service name for Bluetooth Classic, where <*string*> can vary from 1 to 16 alphanumeric characters.

Default:	SerialPort	
Example:	`SS,SPP`	// Service name set to “SPP”

2.3.22 ST,<H16>,<H16>,<H16>,<H16>

This command *only* applies to BLE.

This command sets the connection parameters for BLE connection. These connection parameters are used for establishing a connection when RN4678 acts as a Master. Connection; parameters can be modified by Action command `T` after the connection is established.

This command expects four 16-bit hexadecimal parameters: minimum connection interval, maximum connection interval, slave latency and supervision timeout, respectively. The unit for the first two parameters is 1.25 ms. The unit for the last parameter is 10 ms.

Default:	0008,001C,0000,0200	

Example:	`ST,0010,0020,0000,0100`	// Set minimum connection interval to
		// be 20 ms, maximum connection
		// interval to be 40 ms, slave latency to
		// be 0 and supervision timeout to be
		// 2.56 seconds

2.3.23 STO,<H8>

Command STO sets the power on stand by timeout. The timeout is in the units of 10.24 seconds. The module will go to Deep Sleep mode after the timeout and consumes least possible current. Pulling the WAKE_UP PIN low will bring the module out of this Deep Sleep mode.

Default: FF

Example: `STO 2` // Sets the power on standby timeout to 20.48 seconds

Note: It is not advised to configure it to 0.

2.3.24 STD, <H8>

Command STD sets the disconnect stand by timeout. The timeout is in the units of 10.24 seconds. The module will go to Deep Sleep mode after the timeout from an disconnect event. In this mode the module consumes the least possible current. Pulling the WAKE_UP PIN low will bring the module out of this Deep Sleep mode.

Default: FF

Example: `STO 2` // Sets the power on standby timeout to 20.48 seconds

Note: It is not advised to configure it to 0.

2.3.25 SU,<H8>

Command SU sets the UART baud rate. Table 2-6 shows the input parameters and their corresponding baud rates.

TABLE 2-6: UART BAUD RATES

Baud Rate Index	UART Baud Rate
03	115200
04	57600
05	38400
06	28800
07	19200
08	14400
09	9600
0A	4800
0B	2400
10	307200
11	1000000

Default: 03

Example: `SU,07` // Set the UART baud rate to 19200

© 2016-2019 Microchip Technology Inc.

2.3.26 SW,<H16>

This command *only* applies to Bluetooth Classic.

This command enables low-power Sniff mode which provides low-power operation. In this mode, the device goes into a deep sleep and wakes up every 625 µs x <*value*> to send/receive data. For example, the `SW,0050` setting (0x50 = 80, 80 x625 µs = 50 ms) causes the module to enter low-power sleep and wake once in every 50 ms to check for RF activity.

This setting is useful for applications in which the device is connected and sending data. Data is *not* lost; however, some delay may occur.

Default:	0000	// Disable Sniff mode
Example:	`SW,0050`	// Enable Sniff mode with interval time of 50 ms

2.3.27 SX,<H8>,<H8>

Command `SX` sets the functions of the individual pins available on the module. It expects two 1-byte input parameters. The first parameter indicates the pin and the second parameter indicates the function to be assigned to the pin. Pins and functions are listed in Table 2-7 and in Table 2-8, respectively.

TABLE 2-7: LIST OF AVAILABLE PINS TO ASSIGN FUNCTION

Pin Number	Associated Pin	Default Function
00	P00	UART RTS (Avoid to Change)
01	P05	Null
02	P17	UART CTS (Avoid to Change)
03	P31	Factory Reset
04	P32	Null
05	P33	UART RX Indication
06	P34	Pairing
07	P37	Null

TABLE 2-8: LIST OF FUNCTIONS AVAILABLE TO ASSIGN TO PINS(1)

Function Number	Associated Function	Input/Output
00	No function	N/A
01	UART RTS pin	Output
02	Battery Low Indicator	Output
03	RSSI	Output
04	UART CTS pin	Input
06	Link Drop	Input (NMI)
07	UART RX Indication	Input (NMI)
08	Pairing	Input (NMI)
09	Inquiry	Input (NMI)
0A	Profile Indicate	Output (NMI)
0B	Command/Data mode switch (high for Data mode; low for Command mode)	Input (NMI)
0C	Factory Reset (flip 5 times in the first 5 seconds after powering up)	Input (NMI)

TABLE 2-8: LIST OF FUNCTIONS AVAILABLE TO ASSIGN TO PINS(1) (CONTINUED)

Function Number	Associated Function	Input/Output

Note 1: For more information on the Function Number 0x00 through 0x0A, refer to the *"RN4678 Bluetooth® Dual Mode Module Data Sheet"* (DS50002519).

Example: `SX,01,0B` // Assign command/data switch pin to P05

2.3.28 SY,<0-4>

This command sets the transmit power of the module.

Default: 3

Example: `SY,1` // Change max RF power output to be 1

2.3.29 SZ,<0-1>

This command *only* applies to BLE.

It controls the BeaconThings® feature of the RN4678. A single digit input parameter is expected. If the input parameter is `1`, then the BeaconThings feature is enabled; if the input parameter is `0`, BeaconThings feature is disabled.

Default: 0

Example: `SZ,1` // Enable BeaconThings feature

2.3.30 S-,<string>

This command sets the serialized friendly name of the device, where <*string*> can be up to 11 alphanumeric characters. This command automatically appends the last two bytes of the Bluetooth MAC address to the name, which is useful for generating a custom name with unique numbering. The corresponding get command for `S-` is GN.

Default: N/A

Example: `S-,MyDevice` // Set name to "MyDevice-ABCD"

2.3.31 S?,<0,1>

This command *only* applies to Bluetooth Classic.

The Role Switch command enables and disables the role switch, where an input parameter value of `0` disables role switch and `1` enables role switch. If the switch is set when a device in Default mode is receiving an incoming connection, the device tries to force a role switch enabling the slave to become the master. This option is useful in situations where the local device sends high-speed data up to the remote host, and can improve performance. While connected to the local device, the connected host may *not* be able to make additional outbound (multi-point) connections.

Default: 0

Example: `S?,1` // Enable role switch

© 2016-2019 Microchip Technology Inc.

2.3.32 S$,<char>

This command sets the configuration detect character string, where <*char*> is a single character. This setting configures the escape character to enter Command mode (`$$$`) to a new character string. Restoring the factory defaults returns the device to use `$$$` as the escape character to enter Command mode.

Default: $

Example: `S$,#` // Set ### as string to go into Command mode

2.4 GET COMMANDS

The Get commands retrieve and display the stored information of the device. Get commands do *not* have a keyword or character, and do *not* accept any parameters.

2.4.1 D

This command displays basic settings such as the address, name, UART settings, security, pin code, bonding, and the remote address.

Example: `D` // Display basic settings

Response:
```
BTA=8CDE52DAB1FC
BTName=RN4678-B1FC
Authen=2
CfgChar=$
StatStr=%,%
DevClass=001F00
TxPower=3
Features=1000
Rem=NONE SET
```

2.4.2 GB

This command returns the Bluetooth address of the device.

Example: `GB` // Display the device Bluetooth address

Response: `8CDE52DAB1FC` //Where 8CDE52DAB1FC is the BT address

2.4.3 GK

This command returns the current connection status of the device. There are three single-digit values returned that are separated by a comma.
The first digit indicates the Bluetooth connection status. Value `0` means *no* connection and value `1` means connected.
The second digit indicates authentication status where the possible values are listed in Table 2-9.

TABLE 2-9: AUTHENTICATION STATUS

Value	Description
0	No iAP or BLE authentication
1	Successful authentication
2	APP open; Bluetooth connection over iAP ready
3	APP closed; *no* data transfer possible
4	On power-up, the authentication co-processor test failed
5	Authentication attempted and failed
6	BLE link authenticated

The third digit indicates the type of connection as shown in Table 2-10.

TABLE 2-10: CONNECTION TYPES

Value	Description
0	SPP
1	iAP
2	BLE

Example:	`GK`	// Display current connection status
Response:	`0,4,0`	//Where 0 - Non Connected 4 - On Power-up, the authentication co-processor test failed 0 - SPP (valid only if the device is connected)

2.4.4 G<*char*>

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

Example:	`GA`	// Return Authentication mode set by // command `SA`
Response:	`2`	/Where 2 - Authenticate method set in the module

2.5 ACTION COMMANDS

Action commands perform actions such as inquiries, connecting and entering/exiting Command mode. Action commands do *not* have any parameters.

2.5.1 $$$

This command causes the device to enter Command mode and display command prompt. The device passes characters as data until it sees this exact sequence.

The escape character to enter Command mode can be changed with the `S$` command.

Example:	`$$$`	// Enter Command mode

2.5.2 ---

This command causes the device to exit Command mode, displaying `END`.

Example:	`---`	// Exit Command mode

2.5.3 +

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

Default:	Off	
Example:	`+`	// Turn local echo on

© 2016-2019 Microchip Technology Inc.

2.5.4 &,<Addr>

This command *only* applies to BLE.

Command `&` assigns a random address to the local device. It accepts one input parameter which is a 6-byte random address. This random address can be a static or a private address. For format of random address, refer to *Bluetooth Core Specification version 4.1, Vol 3, Part C, Section 10.8*. If the device is currently advertising, the advertising address immediately changes to the assigned random address.

Example: `&,DF1234567890` // Set random address to be DF1234567890

2.5.5 &C

This command *only* applies to BLE.

Command `&C` clears the random address and uses the local MAC address for advertisement. If the device is currently advertising, the advertising address immediately changes to the local MAC address.

Example: `&C` // Clear random address and use MAC address

2.5.6 &R

This command *only* applies to BLE.

Command `&R` generates a resolvable random address and assigns it as the current random address. This resolvable random address becomes the output to UART as a response to this command. If the device is currently advertising, the advertising address immediately changes to the new resolvable random address.

Example: `&R` // Automatically generate and assign a resolvable
// random address

2.5.7 B

This command *only* applies to BLE.

Command `B` is used to secure the connection and bond two connected devices. Command `B` is *only* effective if the two devices are already connected. Bonding can be initiated by either the central or the peripheral device.

Once bonded, security materials are saved in both ends of the connection. Therefore, reconnection between bonded devices does *not* require pin code exchange, which reduces the reconnection time.
If the bonded connection is lost for any reason, reconnection does *not* automatically secure the link. To secure the link, another `B` command must be issued. However, command `B` is *only* for securing link other than saving connection information.

Default: *Not* bonded

Example: `B` // Bond with connected peer device

2.5.8 C[,<BTAddr>]

This command *only* applies to Bluetooth Classic. It initiates a Bluetooth Classic connection with a peer device.

It expects an optional input parameter which is the MAC address of the peer device.

If *no* input parameter is provided, RN4678 tries to connect to the last connected device. If *no* prior connection is established, an error message is returned.

Example:

`C,112233445566` // Connect to Bluetooth Classic device with
// MAC address 0x112233445566

2.5.9 C,<0,1>,<address>

This command *only* applies to BLE. It connects the local device to a remote BLE device with <*address*>, where <*address*> is specified in hex format. The first parameter indicates the address type: `0` for public address and `1` for private random address. When this command is used to connect to a device which is already bonded, the link is *not* automatically secured. Use command `B` to secure the link after the connection is established.

Example:

```
C,0,00A053112233          // Connect to the BLE address 00A053112233
```

2.5.10 C<1-8>

This command is used to reconnect to a previously connected device that is stored in the saved paired device list with index from 1 to 8. To display the devices in the saved device list, use the command `Y`. Reconnection with a BLE device with random address may *not* be successful if the target device has changed its address.

Example:

```
C2                        // Connect to the second device in the saved
                          // device list
```

2.5.11 F[,<0-5>[,<hex>]]

This command starts the inquiry process to find Bluetooth devices nearby. By default, the inquiry process lasts 20 seconds, configured by using the command `SL`. Command `F` accepts up to two optional input parameters.

The first optional parameter specifies the Inquiry mode. If the first parameter is *not* provided, normal Inquiry mode is used by default. Depending on the Inquiry mode, a second input parameter may be required. Table 2-11 lists the Inquiry modes and their expected second parameter.

TABLE 2-11: INQUIRY MODES

Mode Index	Mode Name	Mode Description	Expected Second Parameter
0	Normal	Perform normal inquiry; return all available BT classic devices in the neighborhood	None
1	Address	Inquiry results are filtered with a specified BT address	Expects a 7-byte second parameter. The first byte is the address mask and the remaining 6 bytes are the Bluetooth MAC address. The lower 6 bits of the address mask specify if the corresponding byte in the MAC address is effective in filtering. A bit set means that the corresponding address byte is effective.
2	COD	Inquiry results filtered with a specified Class of Device (COD)	Expects a 3-byte COD. Only devices with the same COD appear in the scan result.
3	Device Name	Inquiry results filtered with *only* one specified device name	Expects the device name as the second parameter.
4	RSSI	Inquiry results with RSSI value	None
5	BLE	BLE scan	Optional second 16-bit input parameters for scan window and scan interval

Apart from `F,5` all other commands invoke classic inquiry and only `F,5` triggers the BLE scan.

© 2016-2019 Microchip Technology Inc.

The classic inquiry response to the command 'F' would be in the following format.

<BT Address>,<[Device Name]>,<COD>,<[RSSI]>

The BLE scan response to the command 'F,5' would be in the following format.

<BT Address>,<Address Type>,<[Device Name]>,<RSSI>

Example:

```
F                       // Normal inquiry
F,1,38112233445566      // Inquiry BT address 112233xxxxxx,
                        // where xx means “don’t care”
F,3,ABC                 // Inquiry BT device with name “ABC”
```

2.5.12 IA,<H8>,<Hex>/IB,<H8>,<Hex>/IS,<H8>,<Hex> NA,<H8>,<Hex>/NB,<H8>,<Hex>/NS,<H8>,<Hex>

The set of commands IA/IB/IC and NA/NB/NC sets the advertisement, beacon, and scan response payload format.

All advertisement, beacon or scan response data are composed of one or more Advertisement Structure (AD Structure). Each AD structure has one byte of length, one byte of Advertisement Type (AD Type) and various lengths of Advertisement Data (AD Data). The set of commands either appends an AD structure or removes all AD structures, depending on the first parameter.

Commands starting with letter I make the changes effective immediately without reboot. The changes are *not* saved into NVM and *cannot* survive the power cycle. This command is suitable to broadcast dynamic data in the AD structure.

Commands starting with letter N make permanent changes and are saved into NVM, therefore, command N requires a reboot to take effect.

The second letter in the command indicates the type of information to be changed. Letter A indicates advertisement to be changed; letter B indicates beacon to be changed and letter S indicates scan response to be changed.

The first parameter is the AD type. Bluetooth SIG defines AD types in the Assigned Number list in Core Specification. If the AD type is set to zero, a number *not* defined by the Bluetooth SIG, then all AD structures are cleared. Table 2-12 lists the commonly used AD types.

The second parameter is the AD data. AD data has various lengths and follows the format defined in Bluetooth SIG Supplement to the Bluetooth Core Specification.

TABLE 2-12: LIST OF AD TYPES

AD Type (HEX)	Description
00	Clear all AD structure
01	Flags
02	Incomplete list of 16-bit UUIDs
03	Complete list of 16-bit UUIDs
04	Incomplete list of 32-bit UUIDs
05	Complete list of 32-bit UUIDs
06	Incomplete list of 128-bit UUIDs
07	Complete list of 128-bit UUIDs
08	Shortened local name
09	Complete local name
0A	TX power level
0D	Class of device

TABLE 2-12: LIST OF AD TYPES

AD Type (HEX)	Description
0E	Simple pairing hash
0F	Simple pairing randomizer
10	TK value
11	Security OOB flag
12	Slave connection interval range
14	List of 16-bit service UUIDs
15	List of 128-bit service UUIDs
16	Service data
FF	Manufacture specific data

Example: `IA,09,414243` // Set local name to be "ABC"

2.5.13 JA,<Address>/JA,<0,1>,<Address>

Command `JA` adds a MAC address to the white list, and enables white list feature at the same time. Up to 16 BT classic or BLE devices can be added to the white list.

Standard white list *only* applies to BLE. RN4678 extended the white list feature to Bluetooth Classic. When white list feature is enabled in Master role, *only* devices in the white list will be shown in the inquiry results. When white list feature is enabled in Slave role, *only* devices in the white list can be connected with the local device.

This command expects one or two input parameters. If *only* one input parameter is provided, it indicates the MAC address of a Bluetooth Classic device. If two input parameters are provided, then the first parameter input is a single digit indicating the BLE address type: `0` for public and `1` for private. The second input parameter is the BLE address to be added to the white list. Random addresses that are added to the white list using this method have *no* resolvable capability. Therefore, once a BLE device with a random address changes its address, it is *no* longer included in the white list. For resolvable random address, refer to command `JB`.

Note: Do *not* use this command at the same time with command `JB`. When this command is issued, it automatically turns off the effect of command `JB`.

.

Example: `JA,112233445566` // Add BT Classic device to the white list
`JA,0,010203040506` // Add BLE device with public address
// 0x010203040506 to the white list

2.5.14 JB

This command adds all bonded devices, including both BT Classic and BLE devices, to the white list. If the BLE device listed in the white list has a resolvable random address, then the module has the capability to identify that BLE device even if the resolvable random address has changed to another value.

This command expects *no* input parameter.

Note: Do *not* use this command at the same time with command `JA`. When this command is issued, it automatically turns off the effect of command `JA`.

.

Example: `JB` // Add all bonded devices to the white list

© 2016-2019 Microchip Technology Inc.

2.5.15 JC

This command clears the white list. Once this command is issued, it clears all devices in the white list and disables the white list feature.

Example: `JC` // Clear white list

2.5.16 JD

This command displays all devices that are currently in the white list. If the device is a BT Classic device, *only* the MAC address is shown; if the device is a BLE device, both the address and the address type are shown that are separated by a comma. Address type is a single digit value: `0` for public address and `1` for private address.

Example: `JD` // Display all devices in the white list

2.5.17 K,1

The Kill command disconnects the current link.

Example: `K,1` // Disconnect the current link

2.5.18 O,0

Command `O,0` puts the module immediately into Dormant mode, without any UART response. In this mode, the module enters a Deep Sleep state where there is no RF communication, and the current drawn by the module is the lowest. If the host MCU needs the module to communicate with peer device, the host MCU must force the module out of the Dormant mode by pulling the WAKE_UP pin low. On wake up from Dormant mode, the module starts out of Reset.

Example: `O,0` // Enter low-power Dormant mode

Response: // No response is sent as the module
// Immediately enters the Dormant mode

2.5.19 Q

This command puts the device into Quiet mode which means it is temporarily *not* discoverable or connectable. This command does *not* survive a power cycle or Reset.

The `Q` command settings with different responses are as follows:

- `Q` : The module is undiscoverable for both Bluetooth Classic and BLE
- `Q,0` : The module is discoverable and able to connect for both Bluetooth Classic and BLE. Response is `AOK.`
- `Q,1` : The module is undiscoverable and unable to connect to either Bluetooth Classic nor BLE
- `Q,2` : The module is able to connect, but is undiscoverable in Bluetooth Classic
- `Q,3` : The module is undiscoverable in BLE
- `Q,?` : Displays the current Quiet mode

Example: `Q` // Make the module undiscoverable

2.5.20 R,1

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

Note:	Any changes to the device configuration using the Set commands do *not* take effect until rebooting the device.

Example: `R,1` // Reboot device

2.5.21 T,<H16>,<H16>,<H16>,<H16>

This command *only* applies to BLE.

This command adjusts the connection parameters after establishing the BLE connection.

- The first parameter is a 16-bit hex value for minimum connection interval with unit of 1.25 ms.
- The second parameter is a 16-bit hex value for maximum connection interval with unit of 1.25 ms.
- The third parameter is slave latency.
- The fourth parameter is supervision timeout with unit of 10 ms.

Example:

```
T,0010,0020,0000,0100     // Adjust the connection parameter to be
                          // min_conn_interval: 20 ms
                          // max_conn_interval: 40 ms
                          // slave_latency: 0
                          // supervision timeout: 2.56 seconds
```

2.5.22 U,<Z,1-8>

Command `U` removes one or more devices from the linked device list. It expects one input parameter. The linked device list can be accessed by issuing command `Y`.

If the input parameter is letter `Z`, then all devices are removed from the linked device list.

The input parameter can also be a single digit from 1 to 8, corresponding to any of the eight devices in the linked device list to be removed.

Example: `U,1` // Remove first device from the linked device list

2.5.23 V

This command displays the firmware version.

Example: `V` // Display the firmware version

2.5.24 W

This command is the same as command `Q,0`. It is implemented for backward compatibility.

This command enables discovery and connection after it has been disabled with command `Q`. It reloads the stored value of the Inquiry and Page Window in Inquiry and Page Scan.

Example: `W` // Turn on discovery and enable connectivity

© 2016-2019 Microchip Technology Inc.

2.5.25 X

Command X is used to terminate a scan operation before scan duration expires for both Bluetooth Classic and BLE. It is one of the few commands that does *not* have to wait for a command prompt. It expects *no* input parameter.

Example: X // Stop scanning

2.5.26 Y

Command Y displays the MAC addresses of the devices in the stored device list. The output uses the following format:

```
<index>,<priority>,<BT address>[,<address type>]
```

Where *<address type>* is optional and applies *only* to BLE addresses.

Example: Y // Display devices in the linked device list connection.

2.5.27 Z

Command Z stops current connection attempt before the connection is established. It is one of the few commands that does *not* have to wait for a command prompt. This command does *not* expect any input parameter.

Example: Z // Stop connecting

NOTES:

© 2016-2019 Microchip Technology Inc.

Appendix A. Command Quick Reference Guide

A.1 DEFAULT CONFIGURATION SETTINGS

Table A-1 summarizes all the commands and the default values described in Chapter 2. "Command Reference".

TABLE A-1: COMMAND QUICK REFERENCE GUIDE

Command	Default	Description
Set Commands		
SA,<1-4>	2	Sets the authentication method when a remote device attempts to connect
SC,<H16>(1)	0000	Sets the "Service Class" field in the Class of Device (COD)
SD,<H16>(1)	1F00	Sets the COD last significant word (lsw)
SDM,<TXT>(2)	RN4678	Sets the model string in the BLE Device Information service
SDN,<TXT>(2)	Microchip	Sets the manufacture string
SDR,<TXT>(2)	Current RN4678 software version	Sets the software revision of the firmware
SDS,<TXT>(2)	MAC address of the device	Sets the serial number of the device
SE,<H16>(1)	1101	Sets the UUID
SF,1	N/A	Restores the factory defaults
SG,<0-2>	0	Changes Bluetooth modes
SH,<0,1>	0	Enables low-power operation
SI,<H16>(1)	0100	Sets the length of time the device spends enabling an inquiry scan
SJ,<H16>(1)	0100	Sets the amount of time the device spends enabling page scanning
SL,<H8>	02	Sets the duration of the Bluetooth Classic and BLE scan operation
SM,<value>	0	Sets the mode
SN,<string>	N/A	Sets the device name
SO,<string1>[,<string2>]	%,%	Sets the extended status string
SP,<string>	1234	Sets the security pin code
SQ,<H16>	0	Sets the features of RN4678
SR[,<0,1,Z>]<MAC Address>	N/A	Stores the remote address
SS,<string>	SerialPort	Sets the service name
ST,<H16>,<H16>,<H16>,<H16>(2)	0008,001C,0000,0200	Sets the connection parameters for BLE connection
STO	—	Sets the power on standby timeout
STD	—	Sets the disconnect standby timeout
SU,<H8>	03	Sets the UART baud rate
SW,<H16>(1)	0000	Enables low-power Sniff mode

© 2016-2019 Microchip Technology Inc.

TABLE A-1: COMMAND QUICK REFERENCE GUIDE (CONTINUED)

Command	Default	Description
SX,<H8>,<H8>	listed in Table 2-7	Sets the functions of the individual pins
SY,<0-4>	3	Sets the transmit power
SZ,<0-1>(2)	0	Controls the BeaconThings feature
S-,<string>	N/A	Sets the serialized friendly name of the device
S?,<0,1>(1)	0	Enables and disables the role switch
S$,<char>	$	Sets the configuration detect character string
Get Commands		
D	—	Displays basic settings
GB	—	Displays the device Bluetooth address
GK	—	Displays current connection status
G<char>	—	Displays the stored settings for a Set command
Action Commands		
$$$	—	Enter Command mode
---	—	Exit Command mode
+	Off	Local echo on/off
&,<Addr>(2)	—	Assigns a random address to the local device
&C(2)	—	Clears the random address
&R(2)	—	Generates a resolvable random address
B(2)	Not bonded	Secures the connection and bonds two connected devices
C[,<BTAddr>](1)	—	Initiates a Bluetooth Classic connection
C,<0,1>,<address>(2)	—	Connects the local device to a remote BLE device
C<1-8>	—	Reconnects to a previously connected device
F[,<0-5>[,<hex>]]	—	Inquires Bluetooth devices in the neighborhood
IA,<H8>,<Hex>/IB,<H8>,<Hex>/IS,<H8>,<Hex> NA,<H8>,<Hex>/NB,<H8>,<Hex>/NS,<H8>,<Hex>	—	Sets the advertisement, beacon and scan response payload format
JA,<Address>/JA,<0,1>,<Address>	—	Adds an address to the white list
JB	—	Adds all bonded devices to the white list
JC	—	Clears the white list
JD	—	Displays devices currently in the white list
K,1	—	Disconnects the current link
O,0	—	Puts the device into lowest power consumption mode
Q	—	Puts the device into Quiet mode
R,1	—	Forces a complete device reboot
T,<H16>,<H16>,<H16>,<H16>(2)	—	Adjusts the connection parameters
U,<Z,1-8>	—	Removes the device from the linked device list
V	—	Displays the firmware version
W	—	Turns on discovery and connection
X	—	Terminates a scan operation
Y	—	Displays the MAC addresses of the devices in the stored device list
Z	—	Stops connecting

Note 1: Applies *only* to Bluetooth Classic

2: Applies *only* to BLE

© 2016-2019 Microchip Technology Inc.

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
http://www.microchip.com/support
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Austin, TX
Tel: 512-257-3370

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Novi, MI
Tel: 248-848-4000

Houston, TX
Tel: 281-894-5983

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453
Tel: 317-536-2380

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608
Tel: 951-273-7800

Raleigh, NC
Tel: 919-844-7510

New York, NY
Tel: 631-435-6000

San Jose, CA
Tel: 408-735-9110
Tel: 408-436-4270

Canada - Toronto
Tel: 905-695-1980
Fax: 905-695-2078

ASIA/PACIFIC

Australia - Sydney
Tel: 61-2-9868-6733

China - Beijing
Tel: 86-10-8569-7000

China - Chengdu
Tel: 86-28-8665-5511

China - Chongqing
Tel: 86-23-8980-9588

China - Dongguan
Tel: 86-769-8702-9880

China - Guangzhou
Tel: 86-20-8755-8029

China - Hangzhou
Tel: 86-571-8792-8115

China - Hong Kong SAR
Tel: 852-2943-5100

China - Nanjing
Tel: 86-25-8473-2460

China - Qingdao
Tel: 86-532-8502-7355

China - Shanghai
Tel: 86-21-3326-8000

China - Shenyang
Tel: 86-24-2334-2829

China - Shenzhen
Tel: 86-755-8864-2200

China - Suzhou
Tel: 86-186-6233-1526

China - Wuhan
Tel: 86-27-5980-5300

China - Xian
Tel: 86-29-8833-7252

China - Xiamen
Tel: 86-592-2388138

China - Zhuhai
Tel: 86-756-3210040

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444

India - New Delhi
Tel: 91-11-4160-8631

India - Pune
Tel: 91-20-4121-0141

Japan - Osaka
Tel: 81-6-6152-7160

Japan - Tokyo
Tel: 81-3-6880- 3770

Korea - Daegu
Tel: 82-53-744-4301

Korea - Seoul
Tel: 82-2-554-7200

Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

Malaysia - Penang
Tel: 60-4-227-8870

Philippines - Manila
Tel: 63-2-634-9065

Singapore
Tel: 65-6334-8870

Taiwan - Hsin Chu
Tel: 886-3-577-8366

Taiwan - Kaohsiung
Tel: 886-7-213-7830

Taiwan - Taipei
Tel: 886-2-2508-8600

Thailand - Bangkok
Tel: 66-2-694-1351

Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

Finland - Espoo
Tel: 358-9-4520-820

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Garching
Tel: 49-8931-9700

Germany - Haan
Tel: 49-2129-3766400

Germany - Heilbronn
Tel: 49-7131-67-3636

Germany - Karlsruhe
Tel: 49-721-625370

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Germany - Rosenheim
Tel: 49-8031-354-560

Israel - Ra'anana
Tel: 972-9-744-7705

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Italy - Padova
Tel: 39-049-7625286

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Norway - Trondheim
Tel: 47-7288-4388

Poland - Warsaw
Tel: 48-22-3325737

Romania - Bucharest
Tel: 40-21-407-87-50

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

Sweden - Gothenberg
Tel: 46-31-704-60-40

Sweden - Stockholm
Tel: 46-8-5090-4654

UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

© 2016-2019 Microchip Technology Inc.