Roving Networks Bluetooth™ Product User Manual

This manual contains the programming guide and command set for the complete family of Roving Networks Bluetooth products. All Roving Networks modules, super modules, serial adapters and USB dongle (RN-USB-X) share the same command set. Any commands specific to a given product will be highlighted as they appear in this document. For basic usage information and a summary of common commands refer to the Install Guide for a given product.



Version 4.77
August 27, 2009

Roving Networks reserves the right to make corrections, modifications, and changes to its products, documentation and services at any time, without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Customers are responsible for their products and applications using Roving Networks components. To minimize the risks, customers should provide adequate design and operating safeguards. Roving Networks products are not authorized for use in safety-critical applications (such as life support) where a failure of the Roving Networks product would reasonably be expected to cause severe personal injury or death.

Roving Networks and FireFlyAll are trademarks of Roving Networks, all other trademarks are property of their respective owners.

Copyright © 2009 Roving Networks. All rights reserved.

1 Introduction 3
1.1 Making a Connection 3
2 Modes of Operation 4
3 Configuration 4
3.1 Command Mode vs. Data mode 5
3.2 Local Configuration (via serial port) 5
3.3 Remote Configuration (via Bluetooth) 5
3.4 Using DIP switches and PIO for Configuration 6
4 Command Reference 8
1.2 SET Commands 8
4.1 GET Commands 12
4.2 CHANGE Commands 12
4.3 ACTION Commands 13
4.4 GPIO Commands 15
5 Power Management 16
5.1 Optimizing Inquiry (Discovery) and Page (Connection) Windows 16
5.2 SNIFF mode 17
5.3 Enabling Deep Sleep 17
5.4 Disabling Output Drivers 18
5.5 Lowering Transmit Power 18
6 Profile Settings and Features 19
6.1 Alternate Profile Selection 19
6.2 Using PIO as Modem Control Signals 19
7 Design Concerns 21
7.1 Hardware Signals 21
7.2 Hardware Connections and Power 21
7.3 LED Status 22
7.4 Optimizing for Latency or Throughput 22
7.5 Limitations of 7 Bit Data Mode 22
8 Common Problems and Questions 23
9 Discovery and Connection 24
10 Serial Adapter Configuration 25
10.1 Serial interface hardware flow control 25
10.2 Dip Switch Settings 26
10.3 Instant Cable Replacement 26
Appendix A: Bluetooth Evaluation Board 27
Power Terminals 27
Default LED status 27
Appendix B: Command Quick Reference 30
Appendix C: Factory Default Settings 32
Appendix D: Firmware Revision History 33

1 Introduction

This document contains the software command set and advanced configuration settings for Roving Networks Bluetooth devices. The document covers FireFly and BluePort serial adapters, all Bluetooth modules, and the FirePlug USB dongle. Commands and settings that are specific to a single product or product family will be identified as such in the document.

The reader should have a working knowledge of Bluetooth operation and communications. To program the Roving Networks devices you will need a Bluetooth enabled PC (either built-in or using a USB Bluetooth dongle). Only one device may be programmed at a time. Once programmed and configured device settings remain (independent of power down) until they are explicitly changed or the factory defaults are restored.

Prior to getting started, power on the device and pair it with your PC. Roving Networks Bluetooth devices can be programmed over the Bluetooth link or through the serial interface. To program the device you will need to connect to the device by opening the COM port assigned to the Roving Network device through either the Bluetooth or serial port interface.

Roving Networks Bluetooth devices are programmed with a simple ASCI command language which is similar to the industry standard Hayes AT protocol. **Set** commands configure the module and **get** commands echo the current configuration. Configuration settings modified with the **set** command do not take effect until after the module has been rebooted, even though the get command may show otherwise.

1.1 Making a Connection

By default the Roving Networks device shows up under the Bluetooth Devices directory as Serial Port Profile (SPP) Service “FireFly-ABCD”, where “FireFly” is the type of Roving Networks device and “ABCD” is the last four nibbles of the Bluetoot MAC address. The local device name can be changed. You will need to *pair* with the device by double clicking on the device name and following the menu. The firmware automatically stores up to 8 pairings from remote hosts in a first in, first out fashion.

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. The default mode of Roving modules is an OPEN mode, such that the module does NOT require authentication. However most PC’s will require authentication.

The pass key is a string of alpha or numeric chars, 1 to 16 chars in length. During the initial pairing process, this code is entered on both sides of the Bluetooth connection, and must match to complete the pairing. This passkey is used to create a secure link-key, which is then stored on both devices. Upon subsequent connection attempts, the link-keys are compared and must match before the connection can continue. The default Passkey is “1234”.

To connect to FireFly, browse for services, you should see: “SPP” profile with a virtual COM port. Open this virtual COM port to create a Bluetooth connection. Once connected, the device will be in **data mode** allowing data to flow in both directions as if the serial port were locally attached to the PC. The device must be in **command mode** for configuration and programming. To enter command mode type “$$$” (three

dollar signs) from either the remote Bluetooth connection or the local serial port connection. You must enter command mode within 60 seconds (configurable by setting the config timer).

NOTE: Only one client can make connection to FireFly slave at a time. As a master, it is possible to make multiple connections from FireFly, but only in a point-to-point, serialized fashion. At this time Roving Networks devices do not support multipoint master mode.

2 Modes of Operation

Operational modes can be set using the **SM** command

Slave mode (SM,0)
This is the default mode, whereby other Bluetooth devices can discover and connect to the device. Outbound connections can also be made in this mode.

Master Mode (SM,1)
This mode is useful when the device wants to initiate connections (not receive them). In this mode the device will NOT be discoverable or connectable.

Trigger Master Mode (SM,2)
In this mode, the device will automatically connect to the pre configured remote slave address when a character (or characters) are received on the local UART. Connection will remain open until a configurable idle timer (1 to 255 seconds) expires with no data being received, or a configurable BREAK character is seen.

Auto-connect (Master Mode) (SM,3)
This mode can be set by command, or by setting the external dip switch 3 during power up by the on the FireFly and BluePort serial adapter or setting high PIO6 on the Bluetooth modules. If this mode is set, the device will initiate a connection to the pre-stored remote address immediately upon power up. If no address is stored, an inquiry process will be attempted and the first device found that matches the COD will be stored. In this mode, data is passed without being interpreted by the Blueport (high speed), hence the connection cannot be broken via command. If disconnect occurs, the device will attempt to re-connect until successful.

Auto-connect (DTR Mode) (SM,4)
This mode must be set by command. This mode operates like Auto-Connect Master mode, except that the connection and disconnection are controlled by the external dip switch 3 on the FireFly and BluePort serial adapter and PIO6 on the Bluetooth modules. Setting the Switch ON/PIO6 high will initiate auto-connect process, and OFF/PIO6 low will cause a disconnect.

Auto-connect ANY Mode (SM,5)
This mode must be set by command. This mode operates like Auto-connect DTR mode, except that each time the Switch/PIO is set, an inquiry is performed and the first device found is connected. The stored address is NOT used, and the found address is never stored.

3 Configuration

3.1 Command Mode vs. Data mode

Upon power up the device will be in data mode. To enter command mode, send the characters "$$$" through the serial port or from the remote Bluetooth connection. The device will respond with "CMD". To exit command mode, send "---<cr>". The device will respond with "END".

While in command mode, the device will accept ASCII bytes as commands.

A quick check to see if you are in command mode is to type the "D" and "E" commands after entering command mode. This will show the parameters, such as the *Bluetooth* Name, Class of Device and Serial Port settings. To access configuration, the device must be in command mode by issuing "$$$".

You must enter command mode with in the 60 second configuration window or the module will go into fast data mode where all characters are ignored including the "$$$". If the configuration timer expires while in command mode the device will not go into fast data mode after exiting command mode.

You can not enter command mode from the remote Bluetooth connection if the device is in Master Mode.

3.2 Local Configuration (via serial port)

Use a normal RS-232 pass through cable from PC to send ASCII characters through the terminal to the Roving Networks device. The serial port settings must match the Roving Networks serial port configuration. By default these are set to:

- Baud rate 115,200
- 8 bits
- No Parity
- 1 stop bit
- Hardware flow control enabled

Local configuration works at any time when the device does NOT have a Bluetooth connection, and also works under certain conditions. If the device is in configuration mode and a connection occurs, the device will exit configuration mode, and data will pass back and forth from the remote device.

Run your favorite terminal emulator, HyperTerminal or other program. Type **"$$$"** on your screen. You should see "**CMD**" returned to you. This will verify that your cable and comm. settings are correct. Valid commands will return an "**AOK**", response, and invalid ones will return "**ERR** ". Commands that are not recognized will return a "**?".**

To exit command mode, type **"---"<cr>.** (three minus signs).

NOTE: You can enter command mode locally over the serial port at any time when not connected. Once a connection is made, you can only enter command mode if the config timer has not expired. To remain in configuration mode set the config timer to 255. Also, if the device is in Auto Master mode 3, you will NOT be able to enter command mode when connected over Bluetooth.

3.3 Remote Configuration (via Bluetooth)

It is often useful to be able to perform configuration remotely over a Bluetooth connection. To do this, connect to the device over Bluetooth, and using your terminal emulator, perform the same steps as you would for local configuration above. When finished configuring, be sure to either reset the device, or Send the "---"command, which will exit configuration mode and allow data to pass normally.

NOTE: You can only enter command mode remotely over Bluetooth if you have made a connection and sent the $$$ within the "config timer" window after powerup. This can be modified, the default config timer expires 60 seconds after powerup. Once the timer has expired, any data sent to the device will pass unmodified and unrecognized by the command interpreter. The timer can be set to any value from 0 (disable remote configuration) to 255 decimal , which allows continuous (no timeout) configuration. WARNING: Configuration mode (local or remote) is NEVER enabled when the device is in auto-mode and is connected over Bluetooth.

CONFIG TIMER settings

VALUE (decimal)	DESCRIPTION
0	No remote config, No local config when connected
1-252	Time in seconds from power up to allow config
253	Continous config LOCAL only
254	Contiuous config, REMOTE only
255	Continous config, both LOCAL and REMOTE

3.4 Using DIP switches and PIO for Configuration

Function	DIP Switch (adapters)	PIO (modules)	Settings (OFF = 0VDC / ON = 3VDC)
Factory Reset	1	PIO 4	OFF = disabled, ON = ARMED
Auto Discovery/Pairing	2	PIO 3	OFF = disabled, ON = enabled
Auto-Connect	3	PIO 6	OFF = disabled, ON = enabled
Baudrate	4	PIO 7	OFF = stored setting (115K), ON = 9600

Factory Reset- Set this switch/PIO on power up to arm the reset function. Then toggle OFF / ON three times to reset all settings to the factory defaults (other than the Bluetooth name).

Auto Discovery/Pairing Mode - Used with Switch3/PIO6. If Switch3/PIO6 also set,, the device will perform a device Inquiry Scan, searching for a partner Device with a special matching class, (0x55AA) and once found, store the address of such device into the remote address field, and then auto-connect to the remote device. If Switch3/PIO6 is NOT set, the device will enter slave mode with the special matching class, waiting to be found by the master. This mode is usually set once on both ends of a pair of devices, for instant "cable replacement", and then removed.

Auto Connect mode –This is equivalent to auto-master mode 3 in software. The device will connect to the stored address. If Switch2/PIO3 is also set, new discovery/pairing can be made.

Baud Rate select - used to configure 9600 or software selected (default=115K) baudrate. If the switch is OFF, the stored baudrate setting will be used. When the switch is in the ON position, the baudrate will be set to 9600 regardless of the software setting.

NOTE: for the purpose of configuration above, the swithes/IO pins are sampled ONLY at power up time, (during the first 500milliseconds) so they can be used for other functions once the device is in operation. The exception is the factory reset switch/PIO, which once enabled, can be toggled at any time after powerup, a total of 3 transitions will cause the factory reset to occur.

4 Command Reference

All commands are either one or two characters and can be upper or lower case. Arguments for commands are delimited by a comma. Commands take decimal input except where noted. Text data, such as Bluetooth name, and pin code, are case sensitive. Commands fall into five general categories:

SET COMMANDS	store information to flash, changes take effect after power cycle or reboot
GET COMMANDS	retrieve and display the stored information
CHANGE COMMANDS	temporarily change the value of serial baudrate, parity, etc.
ACTION COMMANDS	perform action such as inquiry, connect, etc.
GPIO COMMANDS	configure and manipulate GPIO signals

1.2 SET Commands

All set commands do not take effect until after the module has been rebooted.

S7,<1,0> 7 bit data mode. 1 to enable, 0 to disable. (setting can be seen with the "d" command).

SA,<1,0> Authentication. 1 to enable, 0 to disable. This will force authentication when any remote device attempts to connect. Regardless of this setting, if a remote device forces authentication, this device will respond with the stored pin code. Once a remote device has exchanged pin codes with this device, a link key will be stored for future use. Up to 8 keys are automatically and permanently in flash on the device, in a first in, first out fashion.

SB,<timer> Send BREAK. This is an immediate command, which can send a BREAK signal on the TX . The timer is used to send a variable length BREAK signal.
Timer value Break length (in milliseconds)
1= 37ms, 2=18.5ms, 3=12ms, 4=9ms, 5= 7ms, 6=6ms.

Example : *"SB,2"* sends a 18.5 millisecond break signal.

SC,<hex word> Service Class (four hex values, 11 used, this is used with Device Class command below to create the 24 bit Class of Device number. Note the service class is interpreted by the inquiring device to determine the service. To see a complete listing of available Bluetooth service classes check the Bluetooth SIG web site.

Example : *"SC,0002"*

SD,<hex word> Device Class (four hex values, major and minor in a 16 bit word, used with service class above)

Example : *"SD,8040"*

To set the Class of Device (COD) to 0x1F0123 use the commands

SC,001F
SD,0123

SE,<1,0> Encryption 1 to enable, 0 to disable.

SF,1 Set Factory Defaults.

SI, <hex word> - Inquiry Scan Window. Sets amount of time device spends enabling inquiry scan (discoverability). Minimum value is 0x0012, corresponding to about 1% duty cycle. Inquiry interval is fixed at 0x800, so time spent in inquiry is 0x12/0x100 by default. Maximum value is 0x800, set to 0x0000 to disable inquiry scan and make device non-discoverable. Default value is 0x0200.

SJ, <hex word> Page Scan Window. Sets amount of time device spends enabling page scan (connectability) . Minimum value is 0x0012, corresponding to about 1% duty cycle. Page Scan interval is fixed at 0x800, so time spent in page scan mode is 0x12/0x800 by default. Maximum value is 0x800, set to 0x0000 to disable page scan and make device non-connectable. Default value is 0x0200.

SL,<E,O,N> Set UART parity. Can be any of, Even, Odd, or None. Only the first character is needed and must be capital.

Example : *"SL,E"* sets the parity to Even.

SM,<5,4,3,2,1,0> Mode (0=slave, 1=master,2=trigger, 3=auto, 4=DTR, 5=ANY)

Example : *"SM,1"* sets the mode to Master

SN,<name> Name of the device, 20 characters maximum.

Example: *"SN,MyDevice"*

S-,<name> Serialized Friendly Name of the device, 15 characters maximum.
This command will automatically append the last 2 bytes of the BT MAC address to the name. Useful for generating a custom name with unique numbering.

Example: *S-,MyDevice* will set the name to "MyDevice-ABCD"

SO,<text> Extended Status String, 8 character maximum. Setting this string to from 1 to 8 characters will enable status messages to be sent to the local serial port. Two status messages are sent, when a Bluetooth connection is established, the string "<text>CONNECT" will be sent. Upon a Disconnect, the string <text>DISCONNECT will be sent. This parameter is useful, for example, when connected to a printer, the printer can examine an escape sequence, if the <text> is set to ESC%, the printer can parse the ESC%CONNECT and ESC%DISCONNECT messages without interfering with normal print jobs. In Trigger or Master modes, the first character of this string is used as the BREAK connection character.

Example: *SO,ESC%*

SP,<text> Security pin code, 20 character maximum. Each time the device success pairs, the BT address will be saved. Up to eight addresses can be stored on a first in first out bases. To erase all stored pairings, reset the passkey command. You can use the same value that is already set.

Example: *SP,secretcode* sets pin code to "secretcode"

SQ,<num> Special configuration commands, num is a decimal number with the following interpretation.

Command	Description
0	Disable all special commands
4	Disable reading the values of GPIO3 and GPIO6 on power-up. This command is used when reprogramming GPIO3 and GPIO6 from their default configuration.
16	Configures the firmware to optimize for low latency data transfers rather than throughput.
128	Allow for fast reconnect. This allows applications to disconnect and reconnect back to back quickly.
256	Set 2 stop bit mode on the UART.

SR,<address> Store remote address, 12 hex digits, (6 bytes) no spaces or characters between digits

Example: *SR,00A053112233* sets the remote Bluetooth address to *00A053112233*

NOTE there are two special characters that can be used for the address parameter:

- SR,Z will erase any stored address.
- SR,I will write the last address seen using the inquiry command. This can be helpful when you just have only one other device in range.

SS,<text> Service Name (1 to 20 characters).

Example: *SS,SerialPort* service name set to "SerialPort"

ST,<number> Configuration timer, number of seconds (range= 0 to 255 decimal) to allow remote configuration over Bluetooth after power up in Slave Mode. In all Master modes, the remote configuration timer is set to 0 (no remote configuration). In Trigger Master Mode, the configuration timer is used as an idle timer to break the connection after time expires with no characters being received.

Examples:

ST,0 disables remote configuration
ST,60 sets remote configuration to 60 seconds (default value)
ST,255 enables remote configuration forever

SU,<rate> Baudrate, {1200, 2400, 4800, 9600, 19.2, 28.8, 38.4, 57.6, 115K, 230K, 460K, 921K }, only the first 2 characters are needed.

Example: *SU,57* sets the baudrate to 57600 baud.

SW,<hex word> Enable low power SNIFF mode. Default is 0000=disabled. SNIFF mode allows extreme low power operation. Device goes into a deep sleep, and wakes up every 625us * <hex word> to send/receive chars.

Example: *SW,0050* enables Sniff mode with interval time of 50 milliseconds

This will cause the module to enter low power sleep, and wake once every 50 milliseconds to check for RF activity. See Section 5.2 for more details on Sniff and managing power.

SX,<1,0> Bonding, if enabled only accept connections from the device that matches the stored Bluetooth address register will be accepted. The stored address register can be set with the SR command or will be set upon the first device pairing.

SZ,<num> Raw baud rate (decimal) allows entering of non-standard baud rates. Based on the formula num = baudrate * 0.004096.

S~,<0,1> Set profile to use according to the table below. See section 6.1 for more details on profiles.

<num>	Profile	Comments
0	SPP	Default , no modem control
1	DUN -DCE	Slave or gateway
2	DUN-DTE	Master or client
3	MDM SPP	With modem control signals
4	SPP and DUN-DCE	Multi-profile

S?,<0,1> Role Switch. Enables and disables Role Switch. If set, when an incoming connection is occurs to a slave mode device, an attempt will be made to force a role switch, allowing the slave to become the master. This is useful in situations where high speed data is being sent from the local device up to the remote host, and can result in better performance. However this may create a situation whereby the connecting host will not be able to make additional outbound connections (multipoint) while connected to this device. Default is DISABLED.

S$,<char> Configuration detect character. This allows a change from the default $$$ to some other character. Factory defaults returns the device to $$$.

S|,<value> Low power connect mode. Disables the Bluetooth radio and LED timers while not connected. When set, the module will cycle between active (discoverable and connectable) and low power deep sleep. This can save considerable power when the module is waiting for long periods of time without a connection. The trade off is additional latency when connecting or pairing. The value is a four digit number

made up of two one byte intervals. The first interval is the OFF period and the second the ON period. Both are in Hex seconds (not decimal). The maximum value is 20 seconds for either of the periods. Default is 0000 always actively waiting for a connection.

Example: S|,2001 // cycle ON for one second and OFF for 32 seconds (HEX 20=Decimal 32).

4.1 GET Commands

D Display basic settings. Address, Name, Uart Settings, Security, Pin code, Bonding, Remote Address.

E Display extended settings: service name, service class, device class, configuration Timer.

M Display remote side modem signal status.

O Display other settings. Configuration character, I/O port values, debug mode.

G<X> Display stored settings for command X. These commands correspond to the SET commands above.

Example: *GS* will return 1 or 0 depending on the value of security.

In addition to the above, there are a few other useful commands available**.**

GB Returns the Bluetooth Address of the device.

GK Returns the current connection status: 1=connected, 0 = not connected.

G& Return a hex byte containing the value of the PIO pins

V Return the software release version

4.2 CHANGE Commands

U,<rate>,<E,O,N> - Temporary UART Change, will change the serial parameters immediately, but not store them. Command will return "AOK" at current settings, then automatically exit command mode, and switch to new baud rate.

Baud rrate must be EXACTLY 4 characters = {1200, 2400, 4800, 9600, 19.2K, 38.4K, 57.6K, 115K, 230K, 460K, 921K }.

Parity is E, O, or N (must be capitals)

This command is effective immediately, does not require reboot.

Example: *U,9600,E* Sets baudrate to 9600, parity even.

4.3 ACTION Commands

$$$ Enter command mode Characters are passed as data until this exact sequence is seen. If any bytes are seen before or after the $$$ characters in a 1 second window, command mode will not be entered and these bytes will be passed on to other side.

NOTE: The device will only enter command mode if it is within the configuration timer window (60 seconds from power up by default). In master mode the configuration timer is set to zero.

The character string to enter command mode is configurable using the *S$* command

Use **---** exit command mode. Exit command mode. "END" will be displayed.

+ Local echo. Toggle local echo of RX chars in command mode. (default is off).

C Attempt to connect to the REMOTE stored address.

C,<address> Connect to the address specified in hex format. The address is also stored as the REMOTE address.

CF<address> Connect and immediately go into FAST data mode. NOTE: you will not be able to enter command mode while connected. PIO6 can still be used to disconnect. Thus PIO6 should be held HIGH before sending this command, as lowering PIO6 will cause a disconnect.

CFI Connect and immediately go into FAST data mode using the LAST address found from the Inquiry command. NOTE: you will not be able to enter command mode while connected. PIO6 can still be used to disconnect.

CFR Connect and immediately go into FAST data mode using the REMOTE address. Similar to the C command but bypasses the configuration timer.

CT<address>,<timer> Connect with TIMER. The device will NOT use or store the remote address, rather will make a connection to the <address> (REQUIRED). The device will automatically disconnect after 7 seconds if no data is seen from UART or BT. An optional timer value can be entered to change the timer. This value is in ¼ seconds. So for a 30 second timer, use 120 as the value. The maximum value is 255 (64 seconds)

F,1 Go into fast data mode, ends configuration immediately.

I<,time>,<COD>	Performs an inquiry scan. Default time is 10 seconds, maximum is 48. COD parameter is optional, 0 or no entry looks for all device classes. When entering a COD you must provide all six characters, i.e. 0040F0 for COD 0x40F0. A maximum of 9 devices will be returned. As devices are found, they are displayed in the format below: <BT address>,<BT name>,<COD> **00A053000123,MySerialPort,72010C**
IN<time>,<cod>	Performs an inquiry scan like the *I* command but does not return the Bluetooth name so it returns much faster since getting the name requires a remote lookup for each device found.
IS<time>	Performs an inquiry scan, with a COD of **0x001F00**, which is the default COD for Roving Networks Serial adapters and modules.
IR<time>	Performs an inquiry scan, with a COD of **0x0055AA**, which is the special COD used By Roving Networks Serial adapters and modules to enable "instant cable replacement".
H	Help, will print out a list of commands and their basic syntax
K,	Kill (disconnect) from the current connection**.** The characters KILL<cr><lf> will be echoed to the local UART once the connection is broken.
L	Link Quality. Returns real-time streaming link quality values at 5Hz. Value returned is a two bytes separated by a comma. A value of "ff" is the highest value. The first byte is the current reading the second byte is the low water mark. Example output: **RSSI =ff,e6**
P,<char>	Passes thru any characters up to a CR or LF while in command mode.
Q	Causes device to be non-discoverable and non-connectable (temporarily). Does not survive a power cycle or reset**. Used with the Z command below. Use the "W" command to re-enable.** This command will return **"Quiet"** as a response**.** To get the lowest power mode, first issue a Q, then a Z. Use the SNIFF settings to get lowest power while connected.
R,1	Forces a complete reboot of the device (similar to a power cycle).
T,<0,1>	Pass receive data (from UART or BT) while in command mode. Returns (T=0 , T=1 based on input).
W	Re-enables discovery and connection. This command reloads the stored value of the Inquiry and Page Window to re-enable. For example, to turn off Discovery but still allow connections, send an **"SI,0000"** command, and follow it with **"W"** command. This command returns **"Wake"** as a response.

Z	Enters low power deep sleep mode (<2ma) when NOT connected. Can only be exited by power cycling the device or toggling the RESET pin on the module (causing a HARD reset). For more on managing power see section 5.
&	Returns the value of the DIP switches on Serial adapters and USB-X dongle, or value of GPIO 3,4,6,7 on other modules.
+	Local echo ON/OFF, when turned on in command mode all characters typed will be echoed to the output. Typing + a second time will turn OFF local echo.

4.4 GPIO Commands

Each GPIO command takes 16 bit parameter made up of 2 bytes, a mask to specify the GPIO number and value for each command. The first byte, the mask, determines which GPIO is affected by the command, and the second byte is the value to set.

PARAMETER[15:0] = MASK[7...0]<<8 || VALUE[7..0]

CMD	VALUE	DESCRIPTION
@	<hexword>	Set direction bits for GPIO
&	<hexword>	Set values for GPIO
%	<hexword>	Store powerup direction bits for GPIO
^	<hexword>	Store powerup values for GPIO
*	<hexword>	Set values for PIO8,9,10,11

There are two registers used to control the GPIO, the first is a direction register. This controls whether the GPIO is an input or an output. The second register is the value to apply to the GPIO if set to an output, or is the value of the built-in weak pull-up resistor if the GPIO is set to an input. These settings are immediate, and do not survive a power cycle.

Examples: S@,8080 sets GPIO-7 to an output
S&,8080 drives GPIO-7 high
S&,8000 drives GPIO-7 low

Power-up values: These 2 registers will apply the direction and values upon each subsequent power-up:
Examples: S%,0101 sets GPIO-0 to an output on power-up
S^,0303 drives GPIO-0 high, and pulls up GPIO-1.

Multiple bits can be set with a single command, any bits with a mask of 0 are left unaffected.

Some GPIO are read at power-up time to perform certain functions, so care must be taken when manipulating them. GPIO3 and GPIO6 are used to automatically set master mode, and auto discovery. If it is desired to use these GPIO for other purposes at power-up, a special command must be used to disable their being sensed at power-up time. This command is "SQ,4<cr>" this will set a flag in a stored register that is read at power-up. The Power-up settings for the GPIO can also be viewed using the "O" (other settings) command.

WARNING: GPIO-4 is used by the system to reset stored parameters to factory defaults. If GPIO4 is pulled high on power-up, and then toggled 3 times, all user settings will return to default values. Therefore this pin should not be used as an output, and should not be driven high at power-up time (first 1 second of operation).

NOTE: GPIO2 and 5 are driven by the embedded software as outputs, they can be disabled using the direction command, (to save power, for example) and used as inputs. If set to outputs the software will override any user values.

SETTING GPIO 8-9-10-11
S*,<hexword> = MASK[11..8] VALUE[11..8]

For the upper 4 GPIO, a single word controls the mask and values, and only the lower 4 bits of each byte are used. The first time this command is used, all 4 GPIO are driven as outputs and remain so until a power cycle. There is no powerup command for these bits, only the interactive one. Some modules do not offer these GPIO.

Examples:
S*,0101 GPIO-8 driven HIGH.
S*,0100 GPIO-8 driven LOW.
S*,0202 GPIO-9 driven HIGH.

5 Power Management

5.1 Optimizing Inquiry (Discovery) and Page (Connection) Windows

In slave mode there are two timers that can be used to lower power of the radio while idle. When not connected, the Bluetooth radio is active for a percentage of time listening for other devices. Other devices can Discovery (inquire) or Connect (page). The amount of time the radio listens is called the window, and the interval at which the process repeats is called the interval.

The inquiry and page interval is fixed at 0x1000 = 2.56 seconds. The default window is 0x200 (320 ms) or 12.5% duty cycle.

By lowering the window values, power can be saved at the expense of possibly missing an inquiry or page request. Since the host usually retries automatically many times, the only downside is a delay in discovery or connection time.

The page scan window can be adjusted using the SJ command. The minimum window for inquiry or page is 0x0012 (11.25ms), this corresponds to about a 0.5% duty cycle.

The inquiry scan window can be adjusted using the SI command. The minimum window for inquiry or page is 0x0012 (11.25ms), this corresponds to about a 0.5% duty cycle.

Thus, average power can be reduced from more than 20ma to less than 5ma in standard mode, and less than 3ma in Sniff mode.

It is also possible (and desirable for security reasons) to completely disable inquiry. Once a host has found and installed a device, inquiry is not needed, only page is used to make a connection. To disable inquiry and still allow connections, set the Inquiry timer to 0 with "SI,0000".

5.2 SNIFF mode

Sniff mode is another power conservation method utilized by Bluetooth. By default, Sniff mode is disabled, and the radio is active continuously when connected (about 25-30ma) . In Sniff mode, the radio wakes up at specific intervals, and sleeps in very low power mode (around 2ma) otherwise. The power savings can be quite dramatic.

To enable it, use the "SW,<hex word>" command.

Example interval timers:
0x0020 = 20ms. (32 decimal * .625 = 20).

0x0050 = 50ms, 0x00A0 = 100ms, 0x0190 = ¼ second, 0x0320 = ½ second, 0x0640 = 1 second.

Sniff mode only pertains to an active connection. When a connection is made, both master and slave must support Sniff mode, and agree to the Sniff window, otherwise the radio will stay in full active on mode. Note: the maximum allowed Sniff interval is about 20 seconds = 0x7FFF sniff word setting.

5.3 Enabling Deep Sleep

Deep Sleep mode can be used to obtain extremely low power operation. The device totally shuts down and only draws about 300uA of current in this mode. To enable Deep Sleep, set the high order bit of the Sniff word = 0x8000. This bit is NOT used to determine the sleep interval, it is only used as a flag to enable deep sleep. For example, If you want ½ second sleep 0x0320, with Deep sleep, you would set the sniff word to 0x8320.

In normal low power sleep (not deep sleep) the firmware is still running in idle mode, and wakes up about 20 times per second to check ports, update Leds etc. During Deep sleep, the firmware actually stops running some tasks. For example, the LEDs only update about once per second.

There are 3 ways to wake the radio from sleep mode. The first is to send a character to the UART. Transitions on the RX pin will wake the device from sleep. Wake time is worst case 5ms. Because of this, the first character sent is generally lost by the radio. A better way to wake the radio is to toggle the CTS line from LOW to HIGH,
Wait 5ms, and then send data. The third way is automatic, the radio will wake every <hex word> slot times (1 slot time = 625us) as defined above. The radio wakes and listens to see if the other side of the connection has anything to send. This wake time is typically about 5ms (8 slots) even if no data is to be transferred.

Once the radio is awake it will stay active for exactly 1 second of inactivity, and then sleep again.

NOTE: setting this mode can cause latency issues, and dropped bytes/loss of performance in cases where large amounts of data are being transferred. The nuances of Bluetooth Sniff can be complex, contact Roving Networks if necessary for more details on how to utilize Sniff mode.

To enable even lower power utilization, use the **S|, <hex word>** command to set an on/off duty cycle.

5.4 Disabling Output Drivers

Use the command " S%,1000" to set all PIO0-11 to inputs.
This will also turn off the LED (PIO5) on the FireFly adapter.

5.5 Lowering Transmit Power

All Roving Networks Bluetooth modules and adapters adhere to the Bluetooth (1.1, 1.2, and 2.0/2.1) specifications for power control. Both the RN-21 and RN-41 radios are Class 1 capable.

The power output of the radio is automatically controlled by the baseband. Depending on the operation mode (inquiry scan, page scan, connected) the power is adjusted. Once a connection is made, the radios on both sides negotiate a power setting based on the perceived signal strength (RSSI).

The transmit power can be controlled to:

- reduce effective range for security reasons
- lower radio emissions for agency compliancy concerns
- reduce total power consumption

Use the SY command can be used to configure the transmit power: Using a command, the radio can be instructed to use a value that is lower than the default.

SY,<hex value> where hex value is unsigned hex representation of the setting.

Setting	Hex value	Approximate output (dBM)
-24	FFE8	-20
-16	FFF0	-10
-12	FFF4	-5
-8	FFF8	0
-4	FFFC	2
0	0001	6
4	0004	12 (default)

If a non-zero value is stored in the variable, then upon power up, the radio will use the HIGHEST value that is **less than or equal to** the stored variable.

Example: if -8 = 0 dBM is desired, you would enter **SY,FFF8**

If 0 = 6dBm is desired, you would enter **SY,0001** (you could also enter 0002, or 0003)

Power setting will take effect after power cycle or reboot or reset pin toggle.

To check the STORED power setting, use the **O** command. you will see **TX Power=1** or other setting equal to the value you entered. To check the current power setting, use the **GY** command. Only the actual value

will be returned. Leading zeroes will not be displayed. The value of GY may not match the value in the O command due how the value is used as described above

6 Profile Settings and Features

6.1 Alternate Profile Selection

The default profile is Serial Port Profile (SPP). The firmware also supports the DUN profile in both master and slave modes.

To change the profile, use the "S~,<num>" command.

<num>	Profile	Comments
0	SPP	Default , no modem control
1	DUN -DCE	Slave or gateway
2	DUN-DTE	Master or client
3	MDM SPP	With modem control signals
4	SPP and DUN-DCE	Multi-profile

The most common use of DUN profile is to enable a BT client to connect to a dialup modem. For this mode, use profile 1 (DUN DCE) via command **"S~,1"** . You may also want to set the Class Of Device so that clients can recognize the device as a Bluetooth modem. The correct COD for a Bluetooth modem is 0x040210. This can be set using the commands below:

"SC,0004"
"SD,0210"

A new option available with version 4.74 firmware allows connection to Either SPP or DUN (but not both at the same time) Both Services are displayed when a host discovers services, and either one can be chosen and connected to. It is also possible to connect using SPP profile, disconnect, and then connect using DUN profile.

6.2 Using PIO as Modem Control Signals

The Firefly and RN-XX modules have the ability to replicate the required modem control hardware signals automatically once a connection is made. These signals are transferred outside the data channel (using RFCOMM control channels) and are automatically updated. The default SPP profile (profile=0) does NOT drive these signals or report back inputs. If DUN or MDM profiles are enabled (profile =1,2, or 3), the following signals are automatically driven and received.

Inputs: (read and sent back over Bluetooth to the remote host), Inputs are ACTIVE LOW.

PIO3 = DCD (switch 2) PIO6 = DSR (switch 3) PIO7 = CTS (switch 4).

Outputs: (sent from the remote Bluetooth host, and driven out)

PIO10 = DTR - this is ACTIVE HIGH PIO11 = RTS - this is ACTIVE LOW.

On the Firefly, the DIP switches can also be used to set/clear the DCD, CTS, and DSR signals. DTR and RTS are available on the 9 pin header as well.

7 Design Concerns

7.1 Hardware Signals

1. RESET circuit. RN-41 contains a 1k pullup to VCC, the polarity of reset on the RN41 is ACTIVE LOW. RN21/22 contain a 1k pulldown, and the reset polarity is ACTIVE HIGH.
A power on reset circuit with delay is OPTIONAL on the reset pin of the module. It should only be required if the input power supply has a very slow ramp, or tends to bounce or have instability on power up. Often a microcontroller or embedded CPU IO is available to generate reset once power is stable. If not, there are many low cost power supervisor chips available, such as MCP810T, MCP102/121, and Torex XC61F.

2. Factory reset PIO4. It is a good idea to connect this pin to a switch, or jumper, or resistor, so it can be accessed. This pin can be used to reset the module to FACTORY DEFAULTS and is often critical in situations where the module has been mis-configured.

3. CONNECTION status. PIO5 is available to drive an LED, and blinks at various speeds to indicate status. PIO2 is an output which directly reflects the connection state, it goes HIGH when connected, and LOW otherwise.

4. Using SPI bus for flash upgrade. While not required, this bus is very useful for configuring advanced parameters of the Bluetooth modules, and is required for upgrading the firmware on modules. The suggested ref-design shows a 6pin header which can be implemented to gain access to this bus. A minimum-mode version could just use the SPI signals (4pins) and pickup ground and VCC from elsewhere on the design.

7.2 Hardware Connections and Power

Placing 3.3Vdc into the PIO's while they are set as outputs will permanently damage the radio modules. The failure mode is short across GND and VCC. Use a 10KΩ resistor in series or a 10KΩ pull up resistor for input and output PIO's respectively.

- Make sure to connect a common ground when using the external TX, RX inputs 0 – 3.3Vdc.
- For a 3 wire DB-9 interface (tx, rx, gnd only) connect/short CTS to RTS, Factory default is hardware flow control enabled CTS and RTS connected.
- When using a 5.0Vdc Input, PIO's require a voltage divider. A good choice is 10K ohm series with 20K to Ground. PIO's are 0-3.3Vdc not 5 volt tolerant.

Hardware Communications Connections for Modules and Eval Board

Radio **TX** → **RX** of the application Micro Controller Unit (MCU)
Radio **RX** ← **TX** of the application Micro Controller Unit (MCU)
Radio **RTS** → **CTS** of the application Micro Controller Unit (MCU)
Radio **CTS** ← **RTS** of the application Micro Controller Unit (MCU)

7.3 LED Status

.

MODE	GREEN LED blink rate
Configuring	**10 times per second**
Startup/Config Timer	**2 times per second**
Discoverable/Inquiring/Idle	**Once per second**
Connected	**Solid ON**

The ***YELLOW*** LED blinks when data is transferred on either the RX or TX pins of the DB9 serial port. It is a physical monitor of the actual voltage, and is not driven by software in the module.

7.4 Optimizing for Latency or Throughput

The firmware automatically must make decisions on when to forward receive data coming into the UART RX input out the RF link. The default mode is to optimize for throughput. In some cases, especially with smaller close spaced incoming data packets, the data can get split, with partial packet forwarded and other data coming later. Unfortunately, Bluetooth has algorithms that can cause significant latency between packets (> 10ms) at certain times. If the host protocol expects data to come as contiguous bursts and has short timeouts, this can cause errors. There is another method of optimization, which will force the radio to attempt to keep small bursts of data together. In this mode we optimize for latency. To enable this mode, there is a special debug command. The command is **"SQ,16".** This sets the latency bit in the firmware. To disable this mode. Use the command **"SQ,0".** The value of the register can be read with the **"GQ"** command, and will read back as HEX, even though it is set as decimal.

7.5 Limitations of 7 Bit Data Mode

Roving Networks firmware supports selectable 7 bit data mode, using the "S7,1" command.

Unfortunately the Bluetooth hardware does not support 7 bit data, so this function is accomplished in the firmware application. While completely functional, the performance in 7 bit mode is less than ideal, because software emulation is required to make this work. Hence, there is a noticeable latency and character per second processing limit in this mode. Therefore it is not recommended that this mode be used if the desired serial baudrate is greater than 9600 baud.

8 Common Problems and Questions

My *Bluetooth* client can see the *FireFly* and its serial service, but I can't connect:

This is most likely caused by a security setting on your client. ***FireFly*** does support authentication by default if the client requires it (**using default pincode of "1234",)** but for ease of use, you may want to turn security off on your client. Some clients have these setting off by default, others have them on. To check and disable security:

From your PC desktop, click My *Bluetooth* Places, go to the *Bluetooth* Device configuration (or Advanced Configuration) drop down menu, click on the client applications tab, Select the *Bluetooth* serial port application name, and click on the properties button, if "secure connection", or "authentication", or "encryption" is checked, un check it.

Changing the clients COM port:

Widcomm stack, the most common stack out there allows you to connect to ***FireFly*** using a "Virtual COM" port mapper. The software installs with a default COM port, usually COM3, COM4, or COM5. To change this setting:

From your PC desktop, click My *Bluetooth* Places, go to the *Bluetooth* Device configuration (or Advanced Configuration) drop down menu, click on the client applications tab, Select the *Bluetooth* serial port application name, and click on the properties button, then you can change the com port.

Connecting to more than one *FireFly* from the same client at the same time:

Bluetooth allows 7 devices at a time in a piconet. The Widcomm stack allows you to create multiple instances of serial port profile and connect to multiple ***FireFlys*** at the same time.
To do this: From your PC desktop, click My *Bluetooth* Places, go to the *Bluetooth* Device configuration (or Advanced Configuration) drop down menu, click on the client applications tab, Select the *Bluetooth* serial port application name, and click on the ADD COM port button, then you can add another *Bluetooth* serial port and assign it to another virtual com port (such as COM9).

9 Discovery and Connection

Example of a Master Discovery/Connection Sequence From power up and no connection:

1) Perform an Inquiry to obtain ***BT_Address*** (unless it is already known).

Sent : $$$ // Places Radio in Command Mode
Reply:CMD<cr>

Sent : I,30**<cr>** // Looks forBluetooth devices
Reply:00A096112233,1F00**<cr>**Inquiry Done**<cr**

2) Store the remote address just found.

Sent : SR,00A096112233 **<cr> (or just SR,I if this was the only device found).**
Reply:AOK**<cr>**

3) Connect.

Sent : C **<cr>** // Places Radio in Connect
Reply:AOK**<cr>**

Device will attempt connection to remote slave. "TRYING" will be displayed.

Reply:<text>CONNECT**<cr>** // this will be displayed once connection is made, if <text> string is defined in the stored parameters.

4) Send /Receive data.

Using PIOs/ Switches for Auto-pairing and Auto-Connection

PIO3 PIO6 Function

low low default slave, no special function.
high low slave, changes COD to 0x0055aa for auto pairing.
low high auto master, uses STORED address..... does not try auto pairing.
high high auto master, auto pairing, looks for first device with 0x55aa stores it, and connects to it.

To enable cable replacement, set PIO3 high on slave, PIO3 + PIO6 high on master. Once paired, PIO3 = low on both devices (so re-pairing with another roving device does not occur)

(PIO3 = DIP Switch 2, PIO6 = Dip Switch 3 on Firefly and Blueport)

10 Serial Adapter Configuration

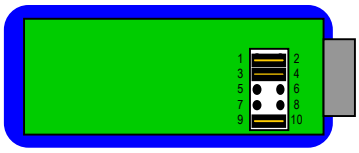
10.1 Serial interface hardware flow control

Pin connections for the Roving Networks FireFly, BluePort and FirePlug products

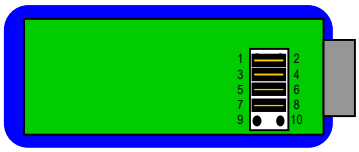
DB9 pin #	FireFly Male	FireFly Female
1	NC	NC
2	RX	TX
3	TX	RX
4	NC	NC
5	GND	GND
6	NC	NC
7	RTS	RTS
8	CTS	CTS
9	V+	V+

FireFly Male DB9

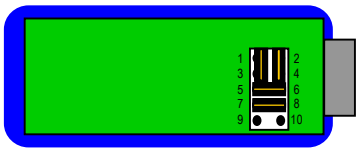
DTE 3 Wire - Flow control DISABLED (Factory Default)
Jumpers
1↔2
3↔4
9↔10



DTE - Flow control ENABLED
Jumpers
1↔2
3↔4
5↔6
7↔8

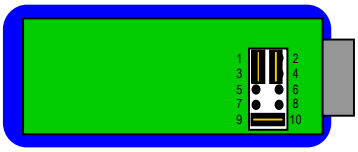


DCE Flow control ENABLED (Num Modem)
Jumpers
1↔3
2↔4
5↔6
7↔8

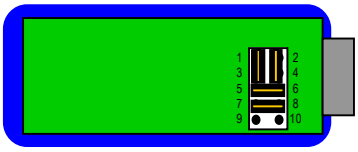


FireFly Female DB9

DTE 3 Wire – Flow control DISABLED (Factory Default)
Jumpers
1↔3
2↔4
9↔10



DTE - Flow control ENABLED
Jumpers
1↔3
2↔4
5↔6
7↔8



10.2 Dip Switch Settings

1 – RESTORE FACTORY DEFAULTS- Set this switch ON, power up unit, and toggle the switch from ON to OFF 3 times to return the unit to factory settings.

2 - AUTO DISCOVERY – In slave mode, sets a special class of device which is used by a remote master to auto connect. If Switch 3 also SET, the device performs a search, stores, and connects to a remote slave which has this switch 2 set.

3 - AUTO MASTER - Device acts as Bluetooth master, auto-connects to a stored remote address. First set the Bluetooth address of the slave device using the SR command or allow the device to auto discover and connect by via Switch 3 AND Switch 2.

4 - DEFAULT BAUD RATE - OFF = 115K will be overridden by software baud rate configuration commands, ON = 9600 ignores any software configuration

10.3 Instant Cable Replacement

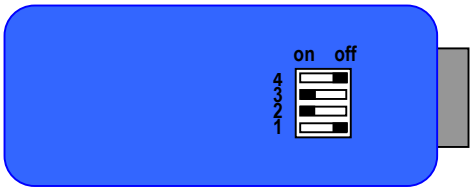
Using two Roving Networks Bluetooth serial adapters or BluePlug USB dongle set the switches on each device as shown below.

Power up both devices and the master will discover the slave device, store its Bluetooth address and connect. The devices are now paired and the green LED on each device should be on solid.

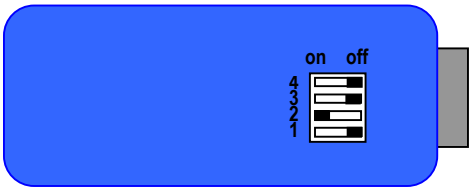
After pairing, set switch 2 on both devices to OFF so that they don't try to re-pair each time power is cycled. Every time the devices get in proximity of one another the will connect. The master will not attempt to connect to any other device.

To break this pairing restore the factor defaults using dip switch 1.

Master mode
Auto discovery and auto Master on



Slave
Auto discovery on

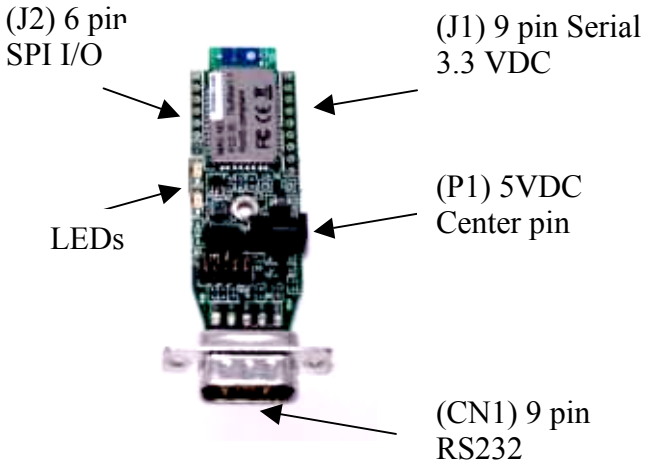


Appendix A: Bluetooth Evaluation Board

The Roving Networks Bluetooth Evaluation board has several interfaces for configuration and programming. These are listed below in the table. The board is powered by and external 5VDC power plug or through pin 9 in the DB9 connector.

The Bluetooth radio same command set and configuration parameters are consistent across all Roving Networks Bluetooth modules, serial adapters and USB dongles so applications can be developed using the evaluation board and run on embedded Bluetooth module such as the RN-41 or RN-24.

Bluetooth Evaluation Board



Power Terminals

Inputs on P1 power connector can be 4.5VDC to 11.0VDC. There is internal regulation down to 3.3VDC for all circuitry. Worst case power draw for the board is 100ma when the *Bluetooth* radio/modem connection is established and transmitting. Power can be as low as 1ma to 25ma average when the *Bluetooth* radio/modem is not connected depending on parameter settings.

Default LED status

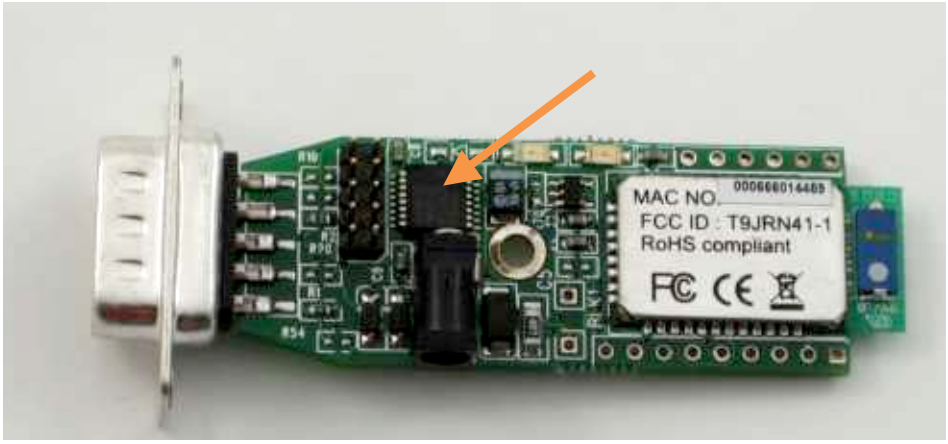
MODE	GREEN LED blink rate
Configuring	**10 times per second**
Startup/Config Timer	**2 times per second**
Discoverable/Inquiring/Idle	**Once per second**
Connected	**Solid ON**

The yellow LED blinks when data is transferred on either the RX or TX pins of the DB9 serial port. It is a physical monitor of the actual voltage, and is not driven by software in the module.

Signal Name	Connector	PIN #	I/O Direction	
Board PWR	P1	1	Power IN (5.0 -9.0 Vdc)	
Board GND	P1	2	<—>	
RS-232 SERIAL-CN1				Modem control options
Pin 1 - DCD	DB9		Not used	PIO2 – BT connect status
Pin 2 - TX	DB9	2	OUT	
Pin 3 - RX	DB9	3	IN	
Pin 4 - DTR	DB9	4	Not used	PIO10- remote DTR out
Pin 5 - GND	DB9	5	<—>	
Pin 6 - DSR	DB9	6	Not used	PIO11- remote RTS out
Pin 7 - RTS	DB9	7	OUT→ * (active low)	
Pin 8 - CTS	DB9	8	IN← * (active low)	
Pin 9 - RING	DB9	9	PWR →IN (4.5 -11Vdc)	
SERIAL 3.3V (J1)				NOTE: the RS232 drive must be removed when using the serial 3.3v signals (see below)
PWR	J1-1	1	Optional 5VDC in	
GND	J1-2	2	GROUND	
CTS	J1-3	3	IN	
RTS	J1-4	4	OUT→ 0 - 3.3Vdc	
TX	J1-5	5	OUT→ 0 - 3.3Vdc	
RX	J1-6	6	IN	
PIO#2	J1-7	7	BT Connection (high state)	
PIO#10	J1-8	8	GPIO	DTR output
PIO#11	J1-9	9	GPIO	RTS output
PIO-SWITCH-LEDS				
PIO#4	SWITCH-1	1	Reset Default Settings	
PIO#3	SWITCH-2	2	Auto Discover and Pairing	DCD input
PIO#6	SWITCH-3	3	Auto Connect as Master	DSR input
PIO#7	SWITCH-4	4	Baudrate (115K - 9600)	CTS input
PIO#5	LED-DL1	GREEN	Pulses for status 0-3.3Vdc	
	LED-DL2	YELLOW	RX, TX data low to high	
PIO#8	LED-DL3	RED	Software controlled	RX char low speed mode
6-Pin SPI (J2)				
MISO	J2-1		Reserved programming)	
MOSI	J2-2		Reserved programming)	
SPICK	J2-3		Reserved programming)	
SPICS	J2-4		Reserved programming)	
PWR	J2-5		Optional 3.3VDC Power	
GND	J2-6		Optional Ground	

Using the Serial 3.3V TTL signals on the evaluation board

The RS232 driver chip must be removed by de-soldering it from the board when connecting to J1 signals. The RS232 chip is the large IC next to the power connector. See below.



Appendix B: Command Quick Reference

SET COMMANDS		FACTORY SETTING
S7,<1,0>	- 7 bit data mode enable/disable	0= disabled
SA,<1,0>	- Authentication enable/disable	0= disabled
SB,<timer>	- Send BREAK	Not Applicable
SC,<hex word>	- Service Class	0x0000= unknown
SD,<hex word>	- Device Class	0x1F00= undefined
SE,<1,0>	- Encryption enable/disable	0=disabled
SF,1	- Factory Defaults	
SI,<hex word>	- Inquiry Scan window	0x0200
SJ,<hex word>	- Page Scan window	0x0200
SL,<E,O,N>	- Parity	N=None
SM,<0,1,2,3,4,5>	- Mode (0=Slave,1=mstr,2=trig, 3=auto, 4=DTR, 5=ANY)	0=Slave
SN,<text>	- Name	FireFly-xxxx
SO,<text> string	- Connect/Disconnect Status String	NULL= no status
SP,<text>	- Pin Code	1234
SR,<adr>	- Remote Address (SR,Z to remove)	NONE SET
SS,<text>	- Service Name	SPP
ST,<num>	- Config Timer	60 seconds
SU,<rate>	- Baudrate	115K
SW,<hex>	- SNIFF rate	0x0000=disabled
SX,<1,0>	- Bonding	0=disabled
S~,<0-4>	- Profile setting 0=SPP, 1=DCE, 2=DTE, 3=MDM, 4=DUN&SPP	0 = SPP
SZ,<num>	- Raw Baudrate	
S?,<0,1>	- Enable /Disable Role Switch	0=disabled

GET, DISPLAY COMMANDS

D	- Basic Settings
E	- Extended Settings
O	- Other Settings
G<X>	- Stored setting
H	- Help
GB	- BT Address
GK	- Connection Status
G&	- I/O Ports
V	- Firmware version

ACTION COMMANDS

+	- Toggle local echo of RX chars in command mode.
C	- Connect immediate to stored remote address.
C,<address>	- Connect to address.
CF<address>	- Connect Fast mode to address.
CFR	- Connect Fast mode to stored remote address.
CT<address>,<timer>	- Connect, required address, optional disconnect timer in ¼ seconds.
F,1	- Enter Fast data mode, end configuration immediate.
L	- Toggle link quality readings.
I,<time>,<cod>	- Device Scan Inquiry, time in seconds, optional cod = class of device filter, 0=all
IN<time>,<cod>	- Device Scan Inquiry, returns NAMEs.
IS<time>	- Device Scan Inquiry, fixed cod=0x001F00 to find Roving devices.
IR<time>	- Device Scan Inquiry, fixed cod =0x0055AA to find instant cable pairs.
K,	- Kill (disconnect) from current connection
Q	- Turn off Discovery and Connectability
R,1	- Reboot
T,<0,1>	- Pass receive data (from uart or BT) while in command mode.
U,<rate>,<E,O,N>	- Temp Uart Change
&	- return the value of the DIP Switches
W	- Re-enable Discovery and Connectability.
Z	- Enter low power Sleep mode

Appendix C: Factory Default Settings

Setting	Default Value
Bluetooth Service Profile	Serial Port Profile (SPP)
Device Mode	0 (Slave)
Baud Rate	115200bps
Parity	None
Data bits	8 bits
Stop bits	1 bit
Power Mode	Auto low power discoverable mode
Name of Device (local name)	FireFly-xxxx (last 2 bytes of BT address)
Service Class	SPP
Service Type	0000 (undefined service type)
Class Of Device	0x1F00 (unknown device type)
Authentication	Disabled
Encryption	Disabled
Discovery Enabled	0x0200 = window, fixed interval of 0x1000= 2.56 Seconds
Connection Enabled	0x0200=window, fixed interval of 0x1000 = 2.56 Seconds
Bonding	Disabled
Config Timer	60 seconds
SNIFF mode	disabled
Default PIN	1234
Local Echo of Rx chars cmd mode	OFF

Note: Restoring factory values resets all values to the above settings except the device name.

Appendix D: Firmware Revision History

Date	Version	
10/6/2006	4.11	Initial production release version on BC04 modules supports 2.0
10/6/2006	4.31	Changed behavior of PIO7/ DIP Switch 4, forces 9600 baud if set, regardless of baud rate set in firmware.
7/1/2008	4.71	New Bluetooth stack with 2.1 support
		Added: command "S-,<name>" allows custom name with serialization
		Added: CT (connect with timer) command
3/7/2009	4.74	Added: support for SPP & DUN simultaneous profile appearance
8/10/2009	4.77	Fixed: disconnect followed by a fast reconnect (< 100ms) would often cause the module to go (deaf) making it undiscoverable or connectable, requiring reset. Fixed: <CR><LF> characters often were sent out the UARTwhen entering fast data mode using the F,1 command remotely. Added: S| command to reduce power while waiting for a connection Added: CF, CFI, CFR commands for fast data mode connect Added: L command to display Link Quality Added: + command to toggle local echo of chars in command mode.