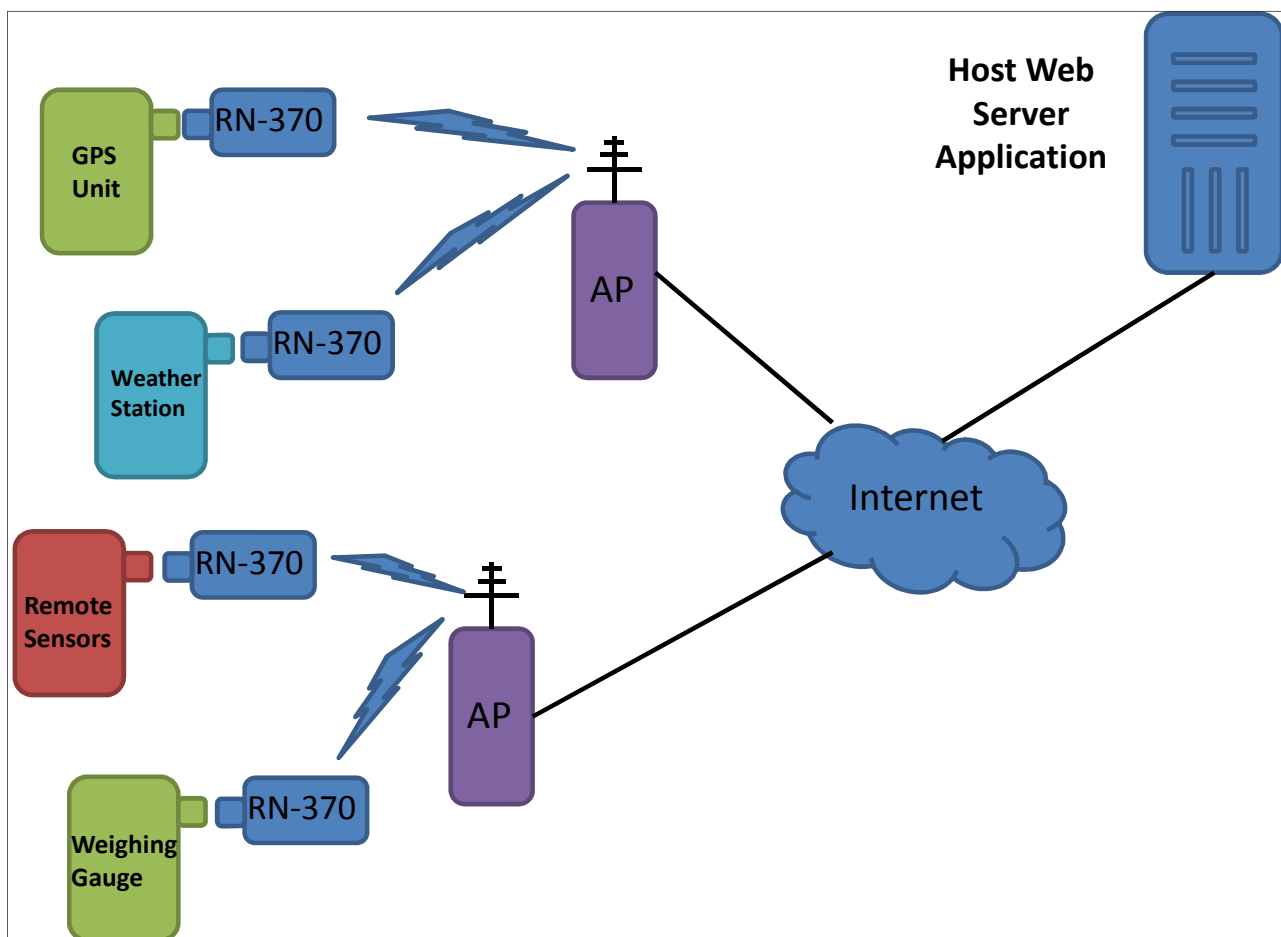


## Overview

The RN-370 is a battery powered RS-232 to Wi-Fi serial adapter. This adapter has a built in HTML client mode which when enabled sends serial data as HTML to a remote Host Web Server. This feature makes it possible to provide Wi-Fi connectivity to applications such as GPS units, remote sensors, weather station, etc.

In this case study, we Wi-Fi enable a weather station using the RN-370 Wi-Fi serial adapter. The weather stations are located anywhere in the world and readings are sent to a central Host Server.

The diagram below illustrates a typical setup. The weather station collects data from different instruments such as thermometer, wind speed monitor, barometer, etc. The RN-370 Serial Adapter is plugged into the RS232 port of the weather station. There are a number of such locations around the world reporting data back to a central Host Server.



The serial adapter is configured with the network name (SSID) and the security passphrase of the wireless access point (AP). The host URL (or IP address) and port number are also configured in the serial adapter.

When the data is ready, the weather station will wake up the Serial Adapter from sleep mode send the data to the remote Host as described below.

The Serial Adapter goes through these steps to send the data to the Host server:

1. Sleep (the Serial Adapter is in sleep mode when it is not sending data to the Host Server)
2. Wake up on RX data
3. Associate to the Network (Access point)
4. Connect out to Host server
5. Send the barometer data to the Host Server
6. Go back to step 1

The RN-370 Wi-Fi serial adapter is powered by two AAA rechargeable batteries. To maximize battery life, it is recommended that the serial adapter be put in sleep mode when it is not transmitting data to the Host serve.

The sleep and the wake periods of the Serial Adapter are customizable and can be tailored to the application. For a detailed explanation on sleep and wake methods, please refer to section 10 in the user manual <http://www.rovingnetworks.com/documents/WiFlyGSX-um.pdf>

**NOTE:** The Serial Adapter is configured to connect out to the Host Server listening at a given port number. The host server accepts incoming requests and responds with a 200 OK.

In this document, we describe the process of configuring the Serial Adapter in HTML client mode. It is assumed that the Host Server is set up and listening on Port 80 (default Web Server port).

## **Configuring the Serial Adapter as HTML Client:**

Configuration is done by placing the RN-370 into command mode and sending ASCII commands.

Configuring the serial adapter involves the following steps:

- Connect the Serial Adapter to a USB-Serial cable.
- Launch a Terminal Emulator program
- Open the COMM port. The default settings of the Serial Adapter are 9600Kbps, 8 bits, No Parity, 1 stop bit.

It is recommend you use TeraTerm as your Terminal Emulator. This is available from the Roving Networks support page [www.rovingnetworks.com/support/teraterm.zip](http://www.rovingnetworks.com/support/teraterm.zip))

## Entering Command Mode

Launch the Terminal Emulator program and make sure that the correct settings are selected (9600kbps, 8 bits, No Parity, 1 stop bit).

Type \$\$\$ into the terminal emulator window (3 dollar signs). You will see a **CMD** returned to you indicating that the RN-370 is now in command mode. Entering a valid command will return an **AOK** and an invalid syntax will return **ERR**.

<CR> returns <x.xx> which is the version of firmware running on the Wi-Fi module.

To exit command mode, enter the command **EXIT**. The module will then exit command mode and return to data mode.

A complete set of commands can be found in section 5 of the Wi-Fi module user guide at <http://www.rovingnetworks.com/documents/WiFlyGSX-um.pdf>

Three sets of parameters need to be configured in the Wi-Fi Serial Adapter to enable the HTML feature. These are: Network, Host Server and Wake up/Sleep parameters. The section below describes these parameters.

### Network Configuration:

set wlan ssid <string>	//sets up network ssid of the AP
set wlan pass <string>	//sets up security passphrase that the AP uses

### Host Server Configuration:

set dns name www.rovingnetworks.com	//set up the URL of the server
set ip host 0	//instructs RN-370 to use DNS address of host server
set ip remote 80	//standard web server port
set ip proto 18	//enable HTTP and TCP protocols
set com remote GET\$/mike.php?ID=	//set up the string

**NOTE:** When HTTP mode is set, the WiFly automatically appends the `\n\n` to the end of the packet. If the html header contains spaces, the \$ is required when entering the string. Space is the command delimiter. When WiFly command parser sees \$ it will convert this to a SPACE character.

**NOTE:** It is not necessary to use DNS. You can connect to the host web server using its IP address. In this case, use the host server's IP address in the "set ip host x.x.x.x" command.

**Wake up, Sleep and Data send configuration:**

```
set sys trigger 0x1           // wake on uart data
set uart mode 0x2             //data trigger mode: make connection on RX data
set sys sleep 10              //sleep after 10 seconds
```

**NOTE#:** When using the wake on uart data mode (**set sys trigger 0x1**), it is possible that the first byte (or possibly multiple bytes) sent to the Wi-Fi serial adapter will be lost, so the designer should take care to send a preamble byte to wake up the module before sending valid data bytes.

A better way to wake the module is to use the CTS input and wait until it is ready to accept data. Wake on CTS can be enabled using the **set sys trigger 0x2** command. When wake on CTS is used, it is not necessary to send a preamble byte.

**Saving setting in configuration file:**

```
save                          //save the configurations in config file
reboot                        //reboot so that the settings take effect
```

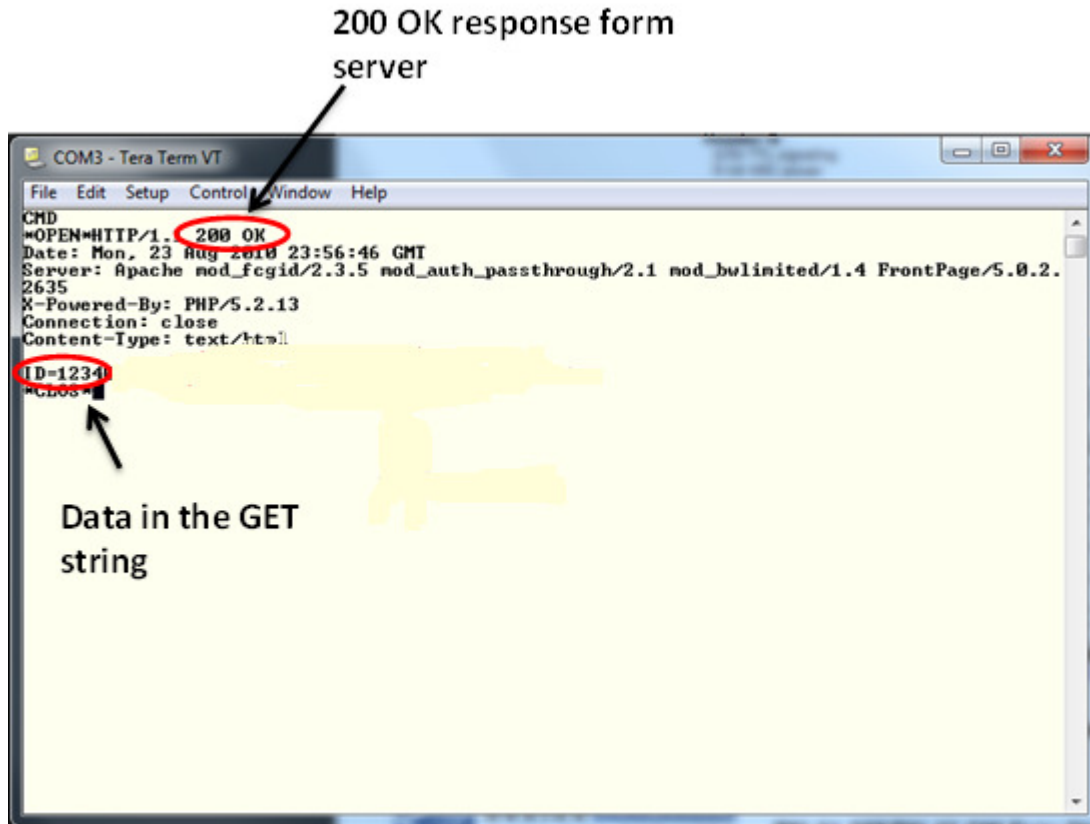
The commands take effect only if they are saved in the config file. After configuring the parameters, save the settings in the config file using the “**save**” command. You will also need to power cycle the module using the “**reboot**” command.

**NOTE:** If you attempt to send data by physically typing characters on the keyboard or if your microcontroller is not sending data fast enough, the WiFly module will send out small packets of data (It will send out many packets of small MTU size). To avoid this, set the flush timer to a higher value (e.g. **set comm. time 5000**).

Packets are also formed based on the flush size. If you set the flush size to 500, then the serial adapter will send out packets of 500 bytes. You can set the flush size by using the **set comm size <value>** command. An IP packet will be sent out each time “value” bytes are received. The maximum value is 1420 bytes.

### Result:

With the above settings, the module will wake up on RX data. It will associate to the Access Point and open connection to the Host Server. It will then send the serial data and then go back to sleep in 10 seconds.



The above diagram shows a screen shot of the HTML data that was sent to the server. As shown, the data in the GET string is appended with the sensor data. The host web server sends 200 OK back to the WiFly serial adapter indicating that it received the data.