Users Manual

HDA800 Evaluation Kit

SPB800S Serial to WiFi solution

H&D Wireless

Revision History

Revision	Revision date	Description
PA1	2010-11-30	First Issue
PA4	2010-12-17	Updated for revision R2B of PC Connection Board
PA5	2011-01-12	Added XPLAIN example
PB1	2011-02-04	Updates for release 1.1.3
PC1	2011-06-09	Updates for release 1.2
PD1	2012-01-10	Updates for release 1.3
PD2	2012-05-02	Updates for release 1.3.1

1 Evaluation Kit Content

- SPB800-E, the SPB800 Evaluation board with a 10 pos. header socket, to allow easy plug in.
- PC connection board
- USB cable
- USB Flash memory with documentation and reference code

2 Product Overview

The SPB800 family of Serial to WiFi modules are a quick and easy way to connect any equipment with a serial port wirelessly to the Internet.

When incorporating the SPB800 into your product development there are two different versions, SPB800S and SPB800P, to be considered depending on your system requirements. Table 2-1 list the differences between the two products.

Feature	SPB800S Serial_to_Wifi	SPB800P oWL-pico
Host	un-aware host	Controlled by host
WiFi parameters	Pre defined through serial interface or web page	Set by host via API
API	None	oWl-pico
FW on host size	0 kB	2 kB
Sockets	1 TCP	8 tcp, 4 udp, 4 raw

Table 2-1: SPB800S vs. SPB800P

The HDA800 development kit gives you a possibility to evaluate both versions before you start your development by changing the FW on the device.

This document mainly describes the SPB800S Serial_to_WiFi, for more information on oWL-pico and download of the latest firmware please visit pico.hd-wireless.se

3 Preparations

To communicate with the SPB800E from a PC you need the PC connection board that convert the SPB800E's UART signal to USB or RS-232 and provide 3.3V power to the module.

On the PC you need a serial communications application, such as TeraTerm or HyperTerm. TeraTerm can be downloaded from http://www.ayera.com/teraterm/download.cfm

3.1 USB driver installation

If you are running Windows 7, it usually recognize the serial to USB chip and installs the drivers at first connection. For other operating system there are driver software provided on the USB memory under the directory called "USB_driver"

If you cannot find the appropriate driver for your system or having other problems with the USB driver please seek more information at http://www.ftdichip.com/Drivers/VCP.htm

Remember to set the two jumpers on the PC connection in their right position to enable USB as serial port.

If you rather use the RS-232 port the jumpers should be in the left position. See Figure 3-1.

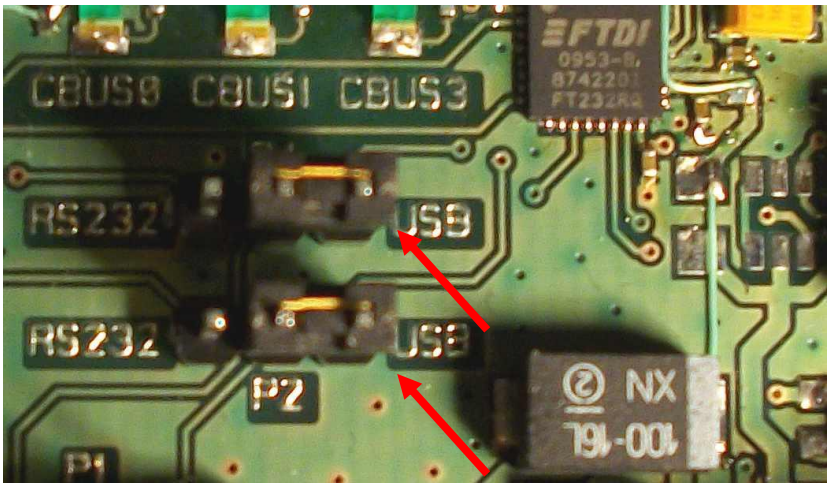


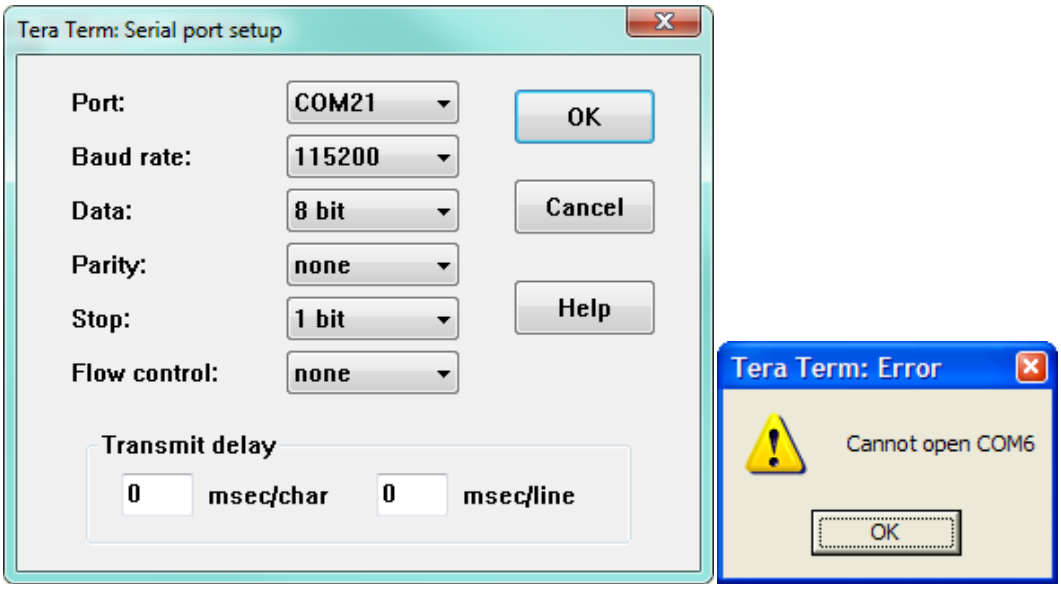
Figure 3-1: Port selection jumpers

4 Connecting the Kit

Connect the Evaluation Kit as follows:

1. Plug in the SPB800E into the 10-pin header.
2. Connect the PC Connection board with the USB cable to a USB port on a PC. The USB powers the board so no other power source is required.
   Or alternatively:
   Power the PC Connection board from an AC/DC adapter with between 5V – 9V DC output and plus on the sleeve and negative on the centre pin. Connect with a serial cable to the PC.

3. Start a serial communications application such as TeraTerm on PC1.
   In TeraTerm, select "Setup->Serial Port…" in the menu and configure the serial port according to the figure below. Note: For firmware releases before 1.3.1 the baud rate should be 57600



   Note that the port number used to communicate with the SPB800E (COM6 in the figure above) might be different in some environments, so make sure to try another one if COM6 fails (see figure above)
   The assigned port can also be found in the Windows Device Manager, under Ports (COM & LPT).

4. Press the reset button on the PC Connection Card to restart the SPB800E

5. In the TeraTerm window a series of dots appears one every second

6. Hit "Return" in the TeraTerm terminal within 5 seconds from the first dot to keep the SPB800E in configuration mode. A $ prompt should show in the terminal.

   Now you are ready to configure the SPB800E for one of the examples

5 SPB800 Firmware

The SPB800 can be programmed with two different firmwares for different applications.

- One, where the host is unaware of the SPB800S, which forwards the data it receives on the serial port onto a TCP socket. The firmware for this mode is called spb800S-serial_to_wifi_rxxx.hlf. This is the firmware on the SPB800 when it is delivered in the HDA800 kit.

- The other, where the host is aware of the SPB800P and uses the oWL-pico API to configure and control the SPB800P, called spb800-pserver_rxxx.hlf, where xxx is the build version corresponding to a release, e.g. 1.2

The firmware can be swapped between the versions the same way as described for upgrades in chapter 8

6 Commands

The following commands are available for the SPB800 in configuration mode. For (o)WL-pico API command please see the (o)WL-pico API Specification.

db	Read/write database
upgrade	fw upgrade xmodem
reset	reset device
help	print a list of available commands
nvdiag	onboard non volatile memory check

The db (data base) command splits in several commands

db reset [path]	reset param(s) to default (need to be followed by a db store)
db get [path]	list parameter(s) in the edit list.
db set <path> [args ...]	set single parameter
db load	read parameters from non-volatile memory (flash)
db store	write parameters to non-volatile memory (flash)

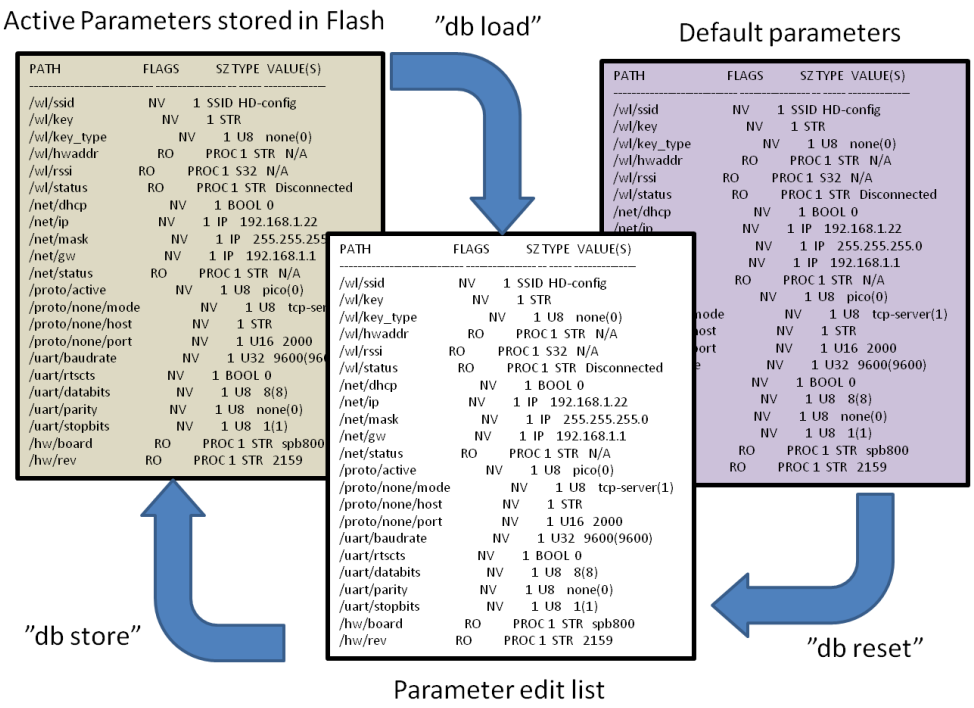


Figure 6-1: Parameter data base

At start up the SPB800 read the parameters from the non-volatile memory, see Figure 6-1. To change a parameter value it needs to be changed in the edit list and then stored in non-volatile memory with the db store command. "db store" stores all the values in the edit list at the same time. The current parameters in non-volatile memory can be retrieved with the command "db load". Factory default setting can be retrieved with the command "db reset". Please note that to restore the active settings to factory default the parameters also have to be stored with "db store".

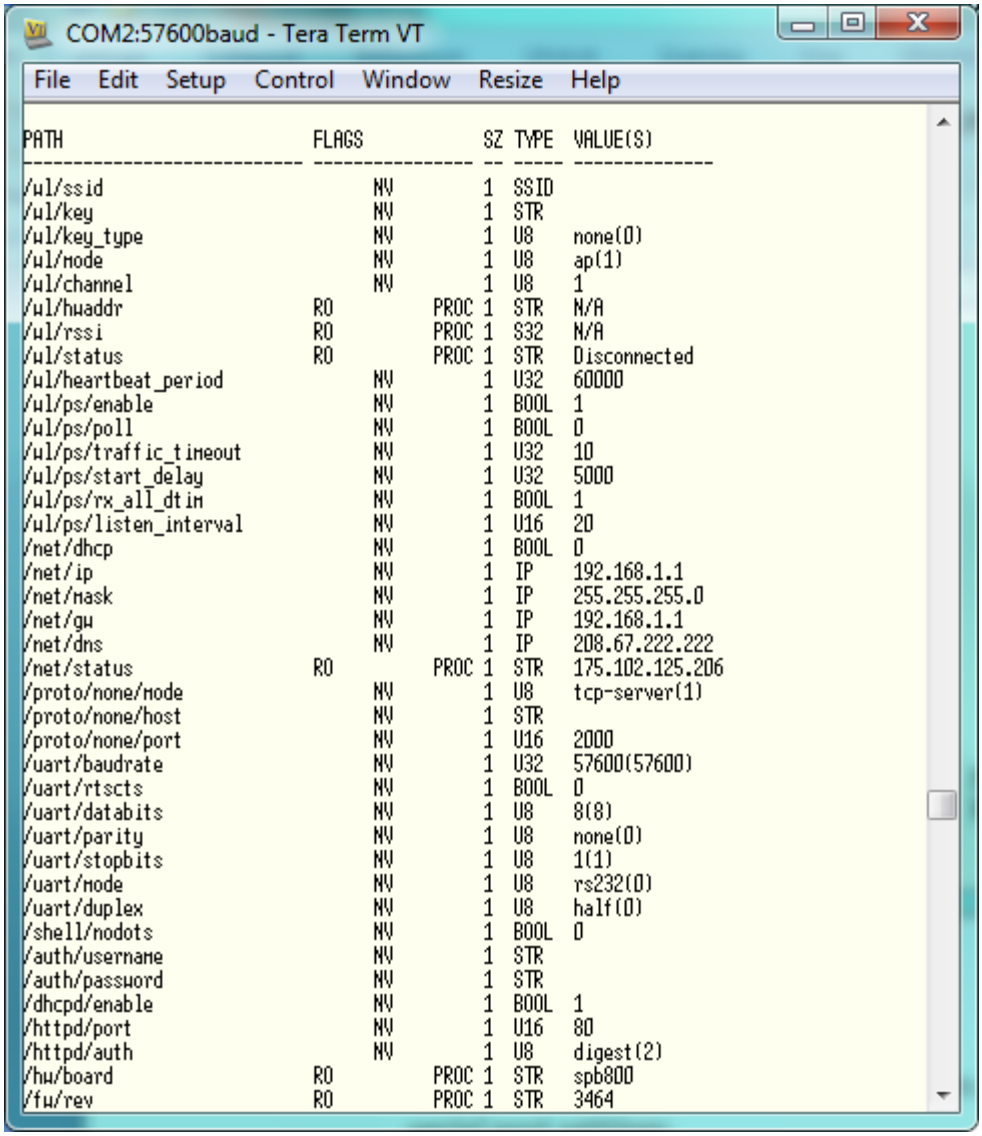


Figure 6-2: Sample of db get listing

- *With "db set" you can set all parameters for the SPB800 to connect to a WiFi Network as well as configure the serial port settings.*
- Typing an erroneous command will list the available command as a help

Table 6-1: Data base parameters

Parameter	**Values**	**Command**
IP Network parameters		
IP address	IPv4 address	db set /net/ip <ip>
Default Gateway	IPv4 address	db set /net/gw <gw>
Net mask	IPv4 address	db set /net/mask <mask>
Enable DHCP client	Boolean	db set /net/dhcp <0 or 1>
DNS server	IPv4 address	db set /net/dns <dns>
Disable DHCP server (AP mode only)	Boolean	db set /dhcpd/enable <0 or 1>
http server port	0 – 65535	db set /httpd/port <80>

Wireless LAN parameters		
SSID	String	db set /wl/ssid <ssid>
Type of key used	none, wep, wpa	db set /wl/key_type <none, wep, or wpa>
WEP, WPA/WPA2 key	String	db set /wl/key <key>
Mode station, AP	sta, ap	db set /wl/mode <sta, ap>
Channel (only in AP mode)	1,2,3,4,5,6,7,8,9,10,11,12,13	db set /wl/channel <1-13>
Heartbeat	ms	db set /wl/heartbeat_period <time>
WLAN Power Save parameters		
Power Save Enable	Boolean	db set /wl/ps/enable <0 or 1>
PS poll	Boolean	db set /wl/ps/poll <0 or 1>
Traffic Timeout	0 -	db set /wl/ps/traffic_timeout
PS Start Delay	Seconds	db set /wl/ps/start_delay
RX all DTIM	Boolean	db set /wl/ps/rx_all_dtim
Listen Interval	1 – 1000 (Beacons)	db set /wl/ps/listen_interval
Serial port parameters		
Baud rate	300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400	db set /uart/baudrate <rate>
Data bits	5, 6, 7 or 8 bit	db set /uart/databits <5,6,7 or 8>
Parity	even, odd, none	db set /uart/parity <even, odd or none>
Stop bits	1, 2	db set /uart/stopbits <1 or 2>
Flow Control	Boolean	db set /uart/rtscts <0 or 1>
RS-232/RS-485	rs232, rs485	db set /uart/mode <rs232, rs485>
UART duplex	half, full	db set /uart/duplex <half,full>
Application parameters		
TCP socket	Port number	db set /proto/none/port <port>
Server/Client	tcp-server, tcp-client	db set /proto/none/mode
Server IP address	String DNS/IP	db set /proto/none/host <ip>
No dots (turn off initial dots)	Boolean	db set /shell/nodots <0,1>
http port	Port number	db set /httpd/port <port>
Security		
Username	String	db set /auth/username <username>
Password	String	db set /auth/password <password>
Http authentication	Basic, digest	db set /httpd/auth <basic,digest>

6.1 Serial Port Commands

All serial ports command is in lower case characters.

6.1.1 db command

The “db” command is used to read, modify and store parameters in the data base. The command expression is

db operation <path> [argument]

Depending on the operation the parameter path and argument may be optional.

6.1.2 db reset <path>

"db reset" resets the parameter indicated in <path> to default. If <path> is omitted all parameters in the edit set is restored to their default values. To restore the unit completely to its default settings the edit set has to be stored in non volatile memory with the command "db store"

6.1.3 db get <path>

"db get" shows the parameter indicated in <path> from the edit list. If <path> is omitted all parameters are listed.

6.1.4 db set <path> [args ...]

"db set" sets the parameter give in <path> to the value given in [args]

6.1.5 db set /wl/mode

"db set /wl/mode (ap|sta) is the setting for if the SPB800 should function as a Soft AP or in Station (client) mode. If this parameter is set there will be an additional delay at reset to change mode.

6.1.6 db load

"db load" retrieve the parameters from the active set to the edit list.

6.1.7 db store

"db store" stores the parameters in the edit list into non-volatile memory.

6.1.8 Power Save parameters

Power save mode is a standardized 802.11 feature to enable the station to turn off its radio to save power. While the station is disconnected the access point will buffer data sent to the station.

- wl/ps/power_save — Enable power save mode
- wl/ps/poll — Use PS-Poll frames to retrieve buffered data. Note: To retrieve one buffered packet, the ps poll scheme needs one ps poll packet to the AP instead of two null packets in the power management bit scheme. Ps poll avoids the overhead of traffic monitoring time in active mode as well. But since each ps poll request can make the AP release only one buffered packet, it is not the optimal scheme for applications with heavy downlink traffic.

- wl/ps/traffic_timeout — Timeout in [ms] to wait for more buffered data from AP. This setting has no effect if use_ps_poll is 1. Any changes to this parameter will take effect immediately.
- wl/ps/start_delay — Power save will delay entering into power save mode with ps_delay [ms], after connecting to an AP. If DHCP client is used the time out should not be shorter than the response time of the DHCP server.
- wl/ps/ rx_all_dtim — If set to 1, then STA will wake up to listen to every beacon containing DTIM (delivery traffic indication messages) when connected. The actual DTIM interval is configured in the AP. If the DTIM interval, as configured in the AP, is larger than the listen_interval, the STA will wake up according to the listen_interval parameter.
- wl/ps/param listen_interval — The Listen Interval field is used to indicate to the AP how often a STA in power save mode wakes to listen to beacon frames. The value of this parameter is expressed in units of Beacon Interval. An AP may use the Listen Interval information in determining the lifetime of frames that it buffers for a STA.

6.1.9 Reset

The “reset” command restarts the SPB800 which loads the parameters from the active set in the data base at start up.

6.1.10 Upgrade

The upgrade command set the device ready to receive new firmware via XMODEM. See Firmware Upgrade page 18 for more information.

6.1.11 Help

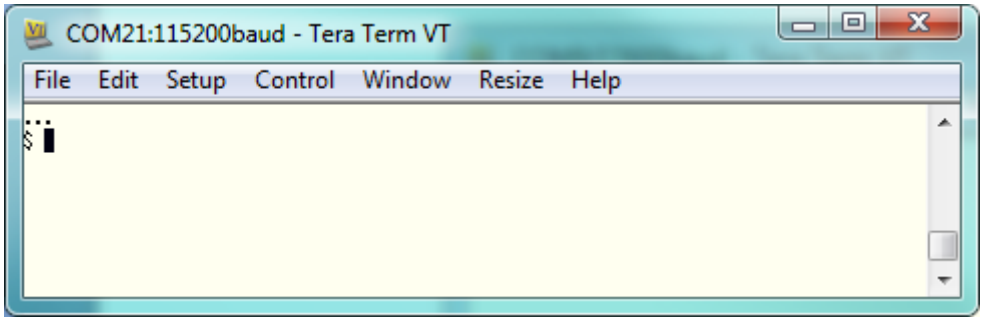
“help” lists the available commands.

7 Example 1, Serial port Wireless LAN adapter

This example describes how to configure the HDA800 Evaluation Kit to a RS-232 – Wireless LAN adapter. It makes use of a direct method changing the parameter database values with commands from a console terminal like TeraTerm or similar. This exemplifies how the SPB800 can be used with an un-aware host.

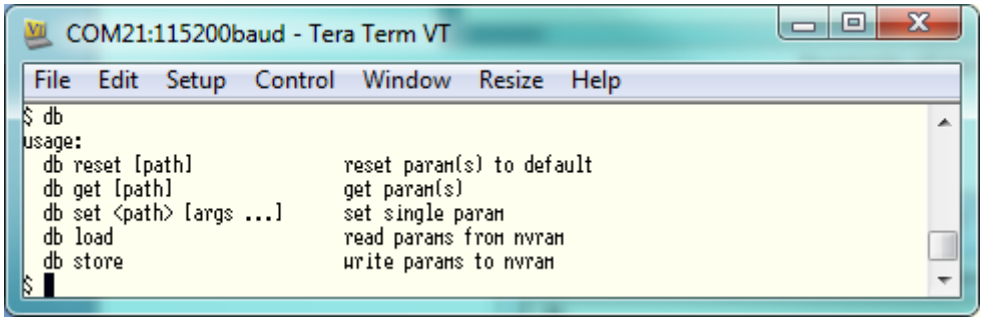
How to do it:

- Start a serial port terminal on the PC
- Connect the HDA800 to the Serial Port/Serial port adapter on the PC
- Insert the DC plug to power the unit and hit "Return" within 5 seconds. You should read a string of dots .. on the terminal and a $ prompt once you have hit return. If you are too late in hitting "Return" the terminal won't do anything.



Example of terminal printout

- The following commands are available



Configuration Example:

- In this example we want to configure the SPB800 to work as an Access point with the following characteristics
  - SSID = my-wifi-net
  - Encryption = WPA
  - Encryption passphrase =my-secret-key
  - IP address we want to assign = 192.168.2.10
  - Subnet mask = 255.255.255.0
  - Default Gateway 192.168.2.254

- Important! No settings are stored in the flash memory until the command "db store" is given.

- First we need to set the parameter /wl/mode to set the mode to Soft AP.

  $ db set /wl/mode ap

- Then we continue with entering the SSID

```
$ db set /wl/ssid my-wifi-net
```

- Then we enter the security encryption type in this case we are using WPA

```
$ db set /wl/key_type wpa
```

- Enter the key

```
$ db set /wl/key my-secret-key
```

- Then we disable the DHCP client as we want to set a static IP address for the SPB800.

```
$ db set /net/dhcp 0
```

- We enter the IP address we have selected for the SPB800. Make sure that is does not collide with any other static set address or the address range managed by the networks DHCP server.

```
$ db set /net/ip 192.168.2.1
```

- We enter the corresponding network mask.

```
$ db set /net/mask 255.255.255.0
```

- Then we enter the default gateway of the network

```
$ db set /net/gw 192.168.2.254
```

- We then in a similar fashion set the RS-232 parameters, first the baud rate to 9600 kbps

```
$ db set /uart/baudrate   9600
```

- We turn flow control off

```
$ db set /uart/rtscts     0
```

- The number of data bits to 8

```
$ db set /uart/databits 8
```

- The number of parity bits to none

```
$ db set /uart/parity none
```

- The number of stop bits to 1

```
$ db set /uart/stopbits 1
```

- Finally we define the SPB800 as server for the TCP socket

```
$ db set /proto/none/mode tcp-server
```

- And set the port of the TCP socket to 2001

```
$ db set /proto/none/port 2001
```

- To check our parameters we type “db get” to list all parameters to check. We need to double check that the parameter /uart/mode is at its default value rs232.

```
$ db get

PATH                        FLAGS     SZ    TYPE    VALUE(S)
--------------------------  ----------  --  -----   ----------
/wl/ssid                    NV        1     SSID    my-wifi-net
/wl/key                     NV        1     STR     my-secret-key
/wl/key_type                NV        1     U8      wpa(2)
/wl/mode                    NV        1     U8      ap(1)
/wl/hwaddr                  RO PROC   1     STR     N/A
/wl/rssi                    RO PROC   1     S32     N/A
/wl/status                  RO PROC   1     STR     Disconnected
/wl/heartbeat_period        NV        1     U32     60000
/wl/ps/enable               NV        1     BOOL    1
/wl/ps/poll                 NV        1     BOOL    0
/wl/ps/traffic_timeout      NV        1     U32     10
/wl/ps/start_delay          NV        1     U32     5000
/wl/ps/rx_all_dtim          NV        1     BOOL    1
/wl/ps/listen_interval      NV        1     U16     20
/net/dhcp                   NV        1     BOOL    0
/net/ip                     NV        1     IP      192.168.2.10
/net/mask                   NV        1     IP      255.255.255.0
/net/gw                     NV        1     IP      192.168.2.254
/net/status                 RO  PROC  1     STR     N/A
/proto/none/mode            NV        1     U8      tcp-server1)
/proto/none/host            NV        1     STR
/proto/none/port            NV        1     U16     2001
/uart/baudrate              NV        1     U32     9600(9600)
/uart/rtscts                NV        1     BOOL    0
/uart/databits              NV        1     U8      8(8)
/uart/parity                NV        1     U8      none(0)
/uart/stopbits              NV        1     U8      1(1)
/uart/mode                  NV        1     U8      rs232(0)
/uart/duplex                NV        1     U8      half(0)
/shell/nodots               NV        1     BOOL    0
/hw/board                   RO  PROC  1     STR     spb800
/fw/rev                     RO  PROC  1     STR     2238
```

- Satisfied with the result we enter the parameters into the flash with ”db store”

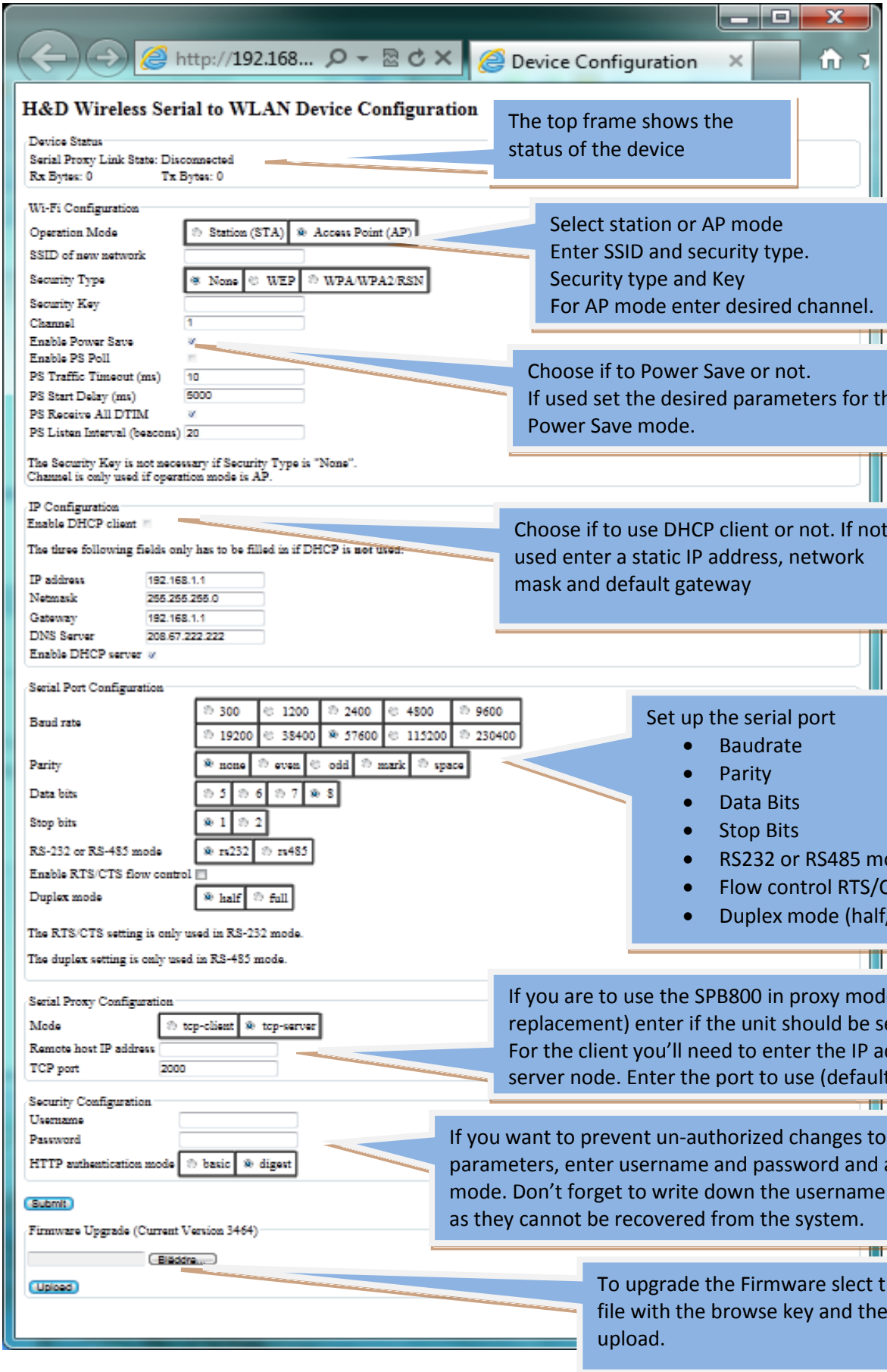
```
$ db store
```

- We can now disconnect the SPB800 from the serial port and DC power and connect it to any equipment that communicates with a RS-232 serial port and access that through a TCP socket on port 2001.

- If the equipment we are to connect the SPB800 to is sensitive for incoming characters it is advisable to turn off the printing of the dots at start up by setting the parameter /shell/nodots to 1. Please note that you will still be able to get into configuration mode the first five seconds even though there are no dots printed.

- See also *1543- HDA800 Quick Start Guide* for more information on this example.

7.1 Web Configuration



To simplify the setup an un-configured SPB800S starts in soft AP mode, with the following properties:

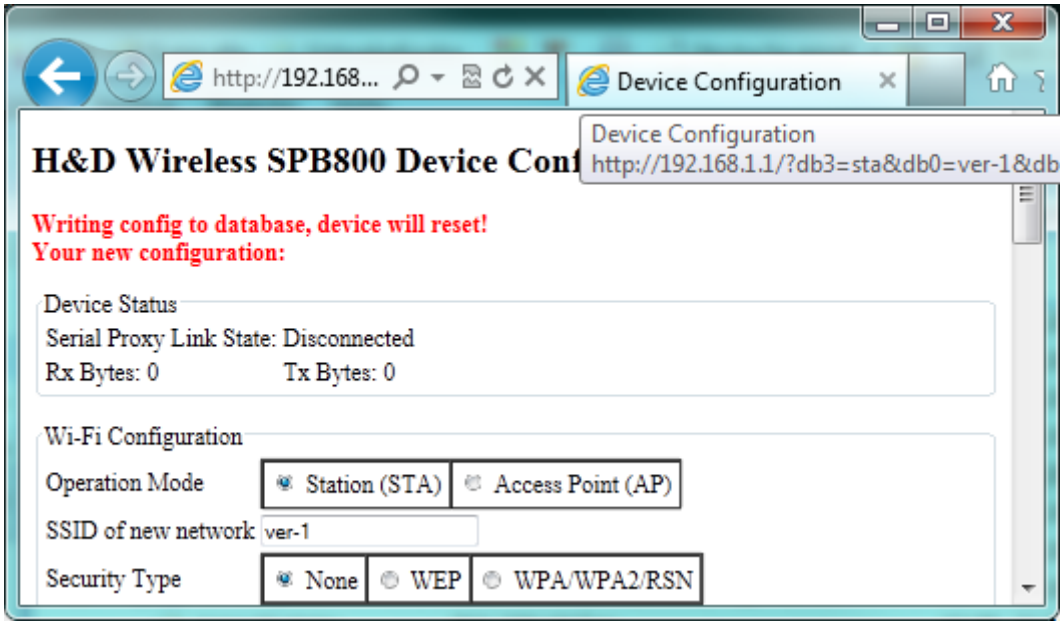
- SSID: hdconfig-xx-xx-xx (where xx-xx-xx are the last six digits of the SPB800's MAC address)
- IP address: 192.168.1.1
- DHCP server

This is the most convenient way to configure the SPB800S as it can be done in situ.

To connect to the SPB800S use a PC or a smart phone with WLAN

- Scan the available networks and connect o hdconfig-xx-xx-xx
- Start a web browser and enter 192.168.1.1 in the address field
- The configuration page opens in the browser
- Enter all the desired data and click on submit.

- A confirmation page with your settings is shown before the SPB800 restart itself with the new settings.
- To change settings or to monitor the unit's activity opens the web page at the new IP address.

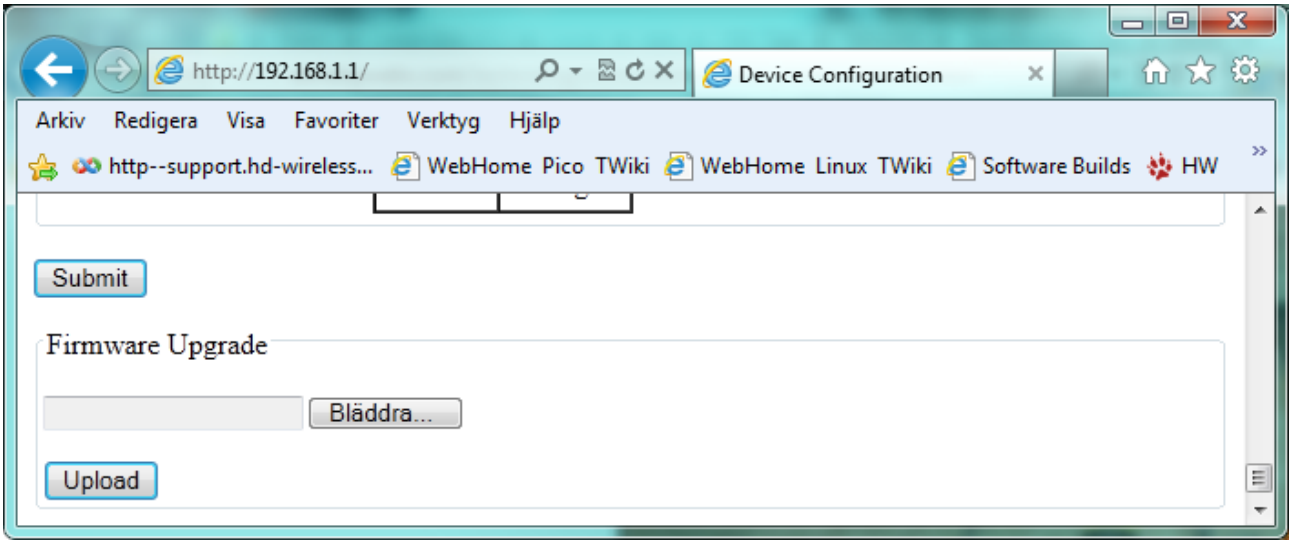


8 Firmware Upgrade

The firmware for SPB800 can be loaded in two ways, via the serial interface or via the WiFi connection.

8.1 Upgrade via WiFi

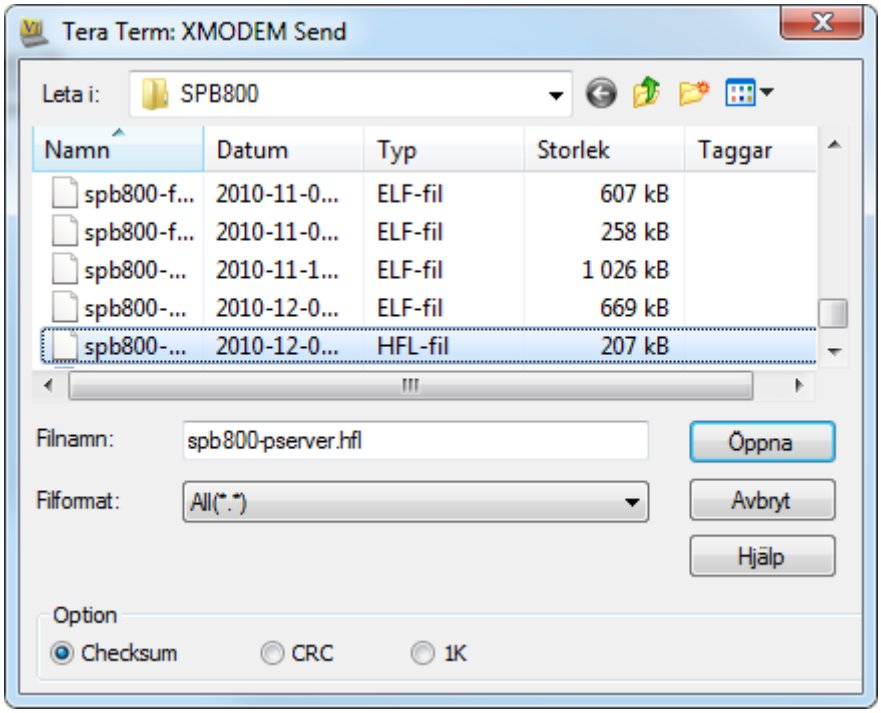
- On the homepage of the SPB800S, scroll to the bottom.



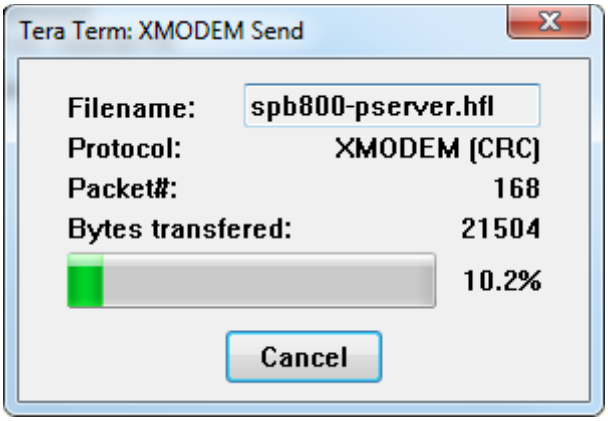
- Click the browse button and select the .hfl file for the firmware you want to load.
- Click upload
- The file will be transferred and the SPB800 will restart with its new firmware.
- Note! Do not disconnect power for the SPB800 or disconnect the PC from the network during upgrade. This may cause the upgrade to fail.

8.2 Upgrade via Serial Port

- Connect the SPB800E with the PC Connection board to a Serial port of a PC and start TeraTerm (115200 8N1)
- Power up or power cycle the SPB800E and hit return in TeraTerm within 5 seconds from the first dot is visible in the terminal window.
- Enter configuration mode by pressing <enter>
- Type "upgrade". The SPB800 will now wait for a file being transferred using the XMODEM protocol.
- In TeraTerm, select File->Transfer->XMODEM ->send and select the firmware image (SPB800-pserver.hfl). Make sure the "checksum" radio button is selected.



- The transfer should start when OK is pressed. Note that SPB800 will time out after about 20 seconds if the file is not sent, in that case restart from typing upgrade and try again.



- When the transfer is complete, the SPB800 will print "completed - rebooting" in the TeraTerm window. Now wait until the dots are printed again.

9 Known Issues

- IP address, mask and GW fields needs to be filled with a valid IP address even if DHCP client is selected.
- Some WLAN cards will not connect to SPB800 acting as an Access Point if they are using Power Save under Windows XP.