

# Loading ISDN / USB custom firmware to a Mitsubishi R100 / ASUS WL500G Wireless Router

[glover.cj@gmail.com](mailto:glover.cj@gmail.com)

Oct 2006

This little guide is designed to take you step by step through the installation of a custom firmware image to the WL500G / R100 (lets call them "R100" – they are identically the same) to connect to the NT1 Plus II ISDN modem via its USB port. The firmware comes with special customisations to handle dialing, authentication and routing of the internet connection. It will automatically redial the connection when it is dropped. It will allow one line to be dropped to allow incoming and outgoing voice calls. The system is designed to allow an always on connection to the internet.

These instructions are only valid for the wl500g / R100 models. Do not follow this method for wl500gd or wl500gp – they have different USB and nvram settings, and are untested.

The system is very stable. It has (and should for you) run fine for months / years without requiring rebooting / maintenance etc.

Note that by following these instructions you probably VOID your R100 device warranty, and there is a possibility that you could 'brick' (ie render useless) your R100. Follow these instructions at your own risk. I make no guarantees.

The process goes like this:

Put the R100 into factory defaults. Load the new firmware. Configure your bigpond username and password. Configure wireless. Enjoy.

## **STEP 1. Initial PC configuration.**

Connect your PC via a network CABLE to a LAN port of the router. Disable the wireless interface on your PC (right click on the wireless icon Disable). Don't try this procedure via a wireless link.

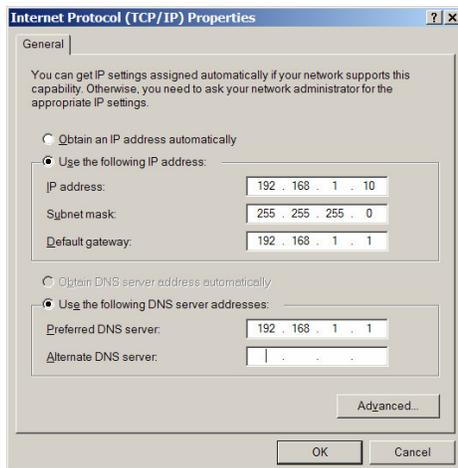
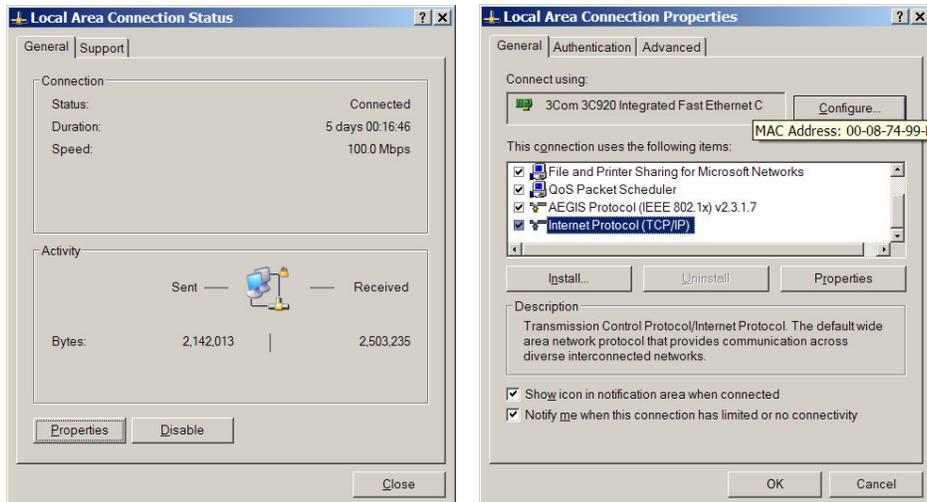
## **STEP 2 Connect to the R100.**

Connect to the R100 router running the factory firmware via the web interface on 192.168.1.1 (username password will be admin admin as per bottom sticker).

Go to system setup, factory default. Press the restore button and wait for the unit to successfully reboot. Log in again and make sure all your previous settings are replaced with the default one. Eg wireless SSID should be "default"

### STEP 3. Set a static IP on the PC.

Set local Ethernet connection on your PC to a static IP on 192.168.1.1 subnet. I chose 192.168.1.10. Do not choose 192.168.1.1 here for the IP!



### STEP 4. Commands for flashing.

Set up the following DOS command windows as shown in the figure below (Start menu, cmd) with the commands. The windows are named 1 – 3 from the top.

DOS window 1: **ping -w 100 -t 192.168.1.1** (then press enter straight away).

DOS window 2: **tftp -i 192.168.1.1 GET ASUSSPACELINK** (do not press enter yet)

DOS window 3: **tftp -i 192.168.1.1 PUT WL500g-1.9.2.7-6b.trx** (do not press enter yet).

*Of course the WL500g-1.9.... will be replaced with the name of the firmware file firmwareR100ISDNauto.trx that you intend to load onto the R100. Make sure you are in the correct directory on your PC; hint: type the first few letters and press TAB – it should auto complete the name.*

*Make sure it's the correct one. Why not create a new directory for this effort with only the firmware in it. Lastly **do not copy and paste the commands into the DOS windows – type them.** Don't even think about asking why, just type them.*

In the first window (I left out the `-w 100` in the screen dump!) you should see ping replies:

```
ping -w 100 -t 192.168.1.1
```

```
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
```

```
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
```

Look at the TTL value of 64. This is normal, it means the router has booted to the firmware. Now we are nearly all setup to flash the firmware.

### **STEP 5. Bootwait**

Start the R100 into bootwait mode.

Remove the power plug to the R100. Once it is off, with a pen, small screwdriver etc, press and hold in the “restore” button on the back of the device. Insert the power plug (with the restore button still held in) and watch the “PWR” LED on the front of the R100. It should light up, then start to flash about once a second. You can stop pressing the restore button now. You should now be in bootwait mode.

Watch the DOS window 1.... “Hardware error” means the network is physically not alive – ie the computer cannot sense the host at the other end. Once the network starts up on the R100, but IP communication is not yet available, the “Destination ....” Error means it could not find 192.168.1.1. Once you get a ping reply look closely at the TTL. If its 128, then you are in boot wait mode. The R100 is waiting for you to load the firmware.

### **STEP 6. Flashing the firmware.**

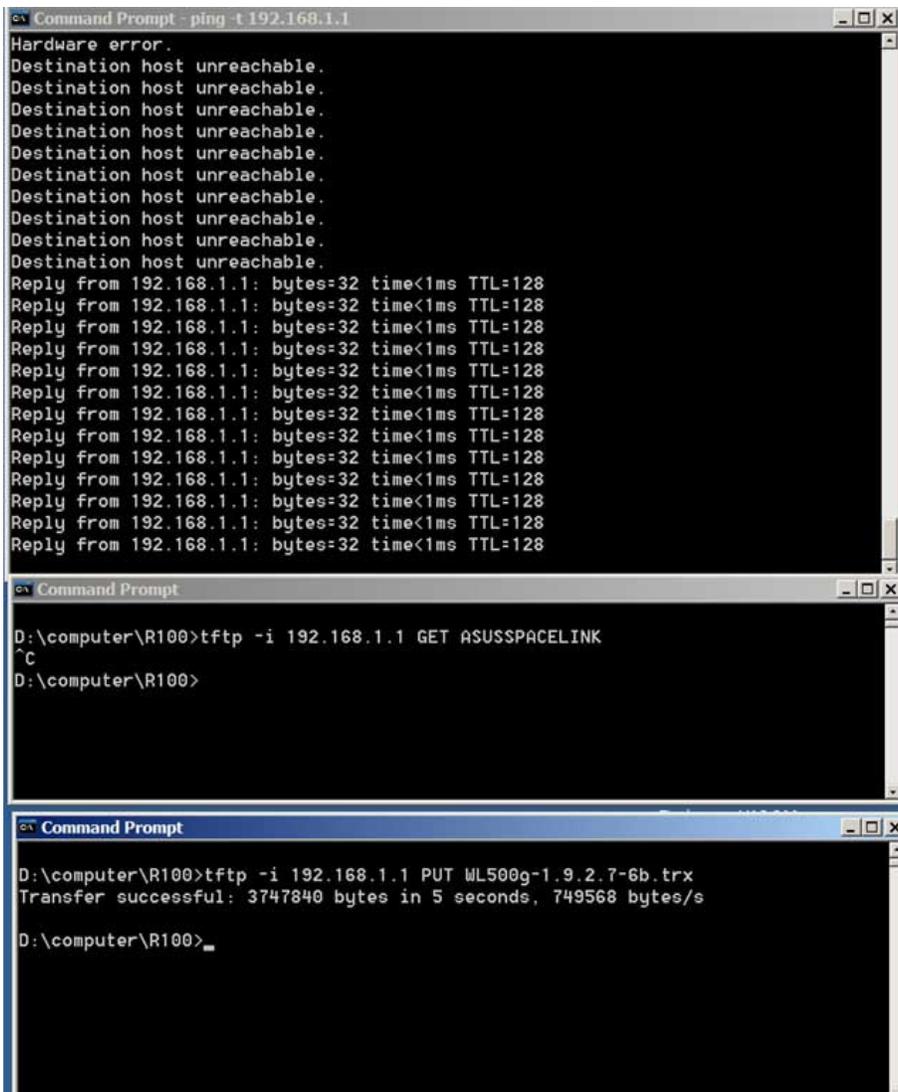
Load the new firmware.

Are you sure you have started getting ping replies as above? And as per Window 1 below? Yes.

Press enter in the second window. If the command does not give an error, press Control C after a second or two. This command tells the R100 that it should expect a firmware will be soon uploaded. (Note that the wl500gd and wl500pg do not need this step). The PWR light now no longer flashes – it stays unlit.

Now press enter in the third window. You have more than several seconds to do this, so don't panic. You should see LAN lights on the router flashing as the file is transferred, and the Transfer succeeded message. **DO NOT TURN ANYTHING OFF OR DISCONNECT ANY CABLES.** Make sure no kids are around at this time in case they accidentally do so (that warning comes from experience).

Leave the router alone for a few minutes. It should reboot by itself. Watch the first window for progress (TTL=64). You can also tell when it has rebooted when the “AIR” lamp lights up and flashes.



## **STEP 7. Rebooting**

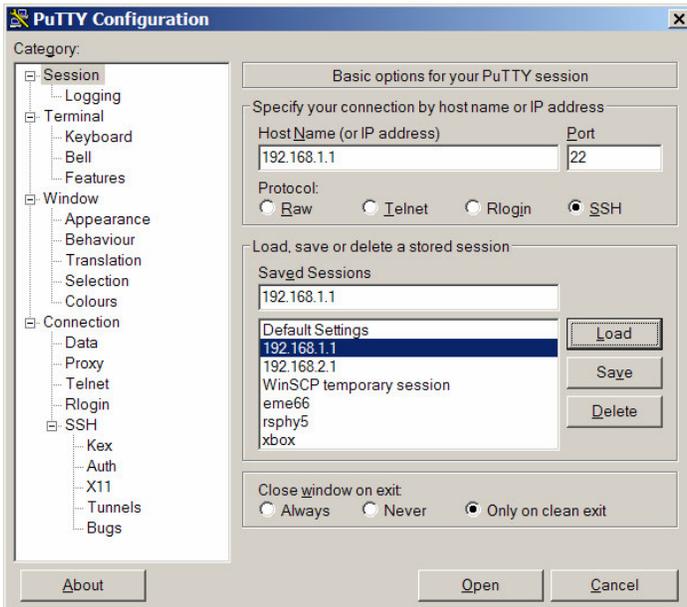
Check that the router has rebooted – watch that the “AIR” light is lit and flashing.

Connect to the web interface again on 192.168.1.1. This time the web interface takes you to the Openwrt Admin Console.

If you have this, then you have successfully loaded openwrt onto the R100. Configuring it comes next.

## **STEP 8. SSH Connect with Putty**

Its time to learn about putty. Putty is a SSH (secure shell) client. It gives you a linux terminal screen on the R100. When you start putty you will get a screen like this. Put 192.168.1.1 in the hostname field, and choose the SSH button. I would save this as a configuration – as you can see I have a few. Click on open.

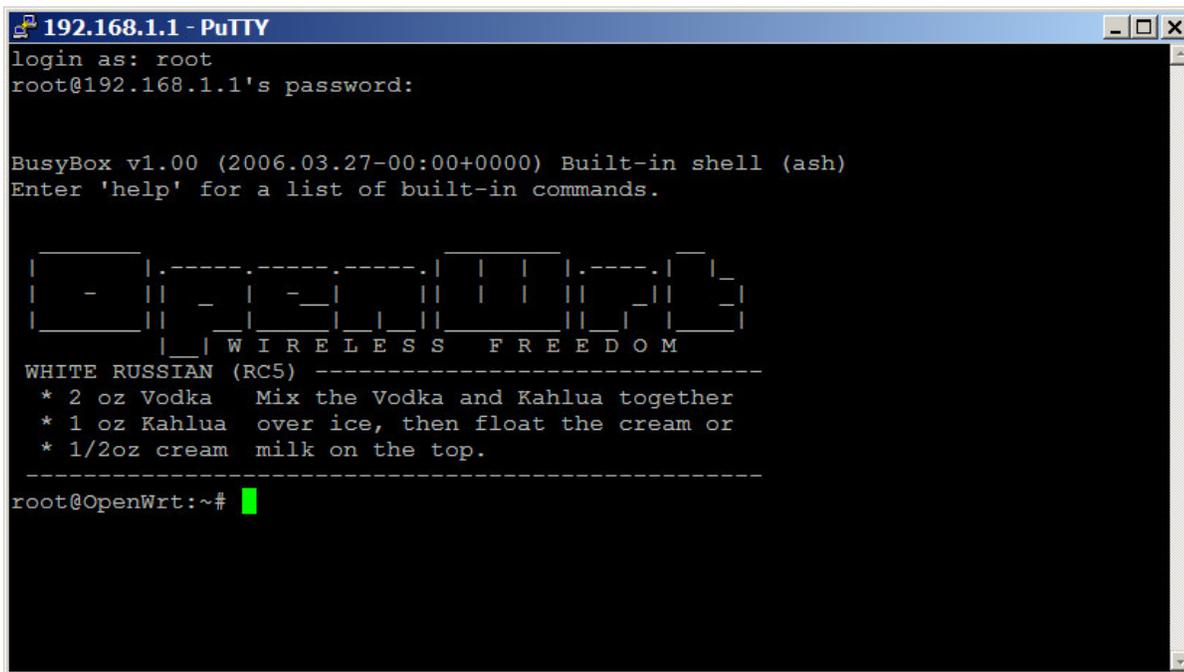


It will give you a terminal screen with:

Logon as: root

root@192.168.1.1's password: admin

ie, Openwrt uses the username / password combo of root / admin. Remember this. This is what you should see.



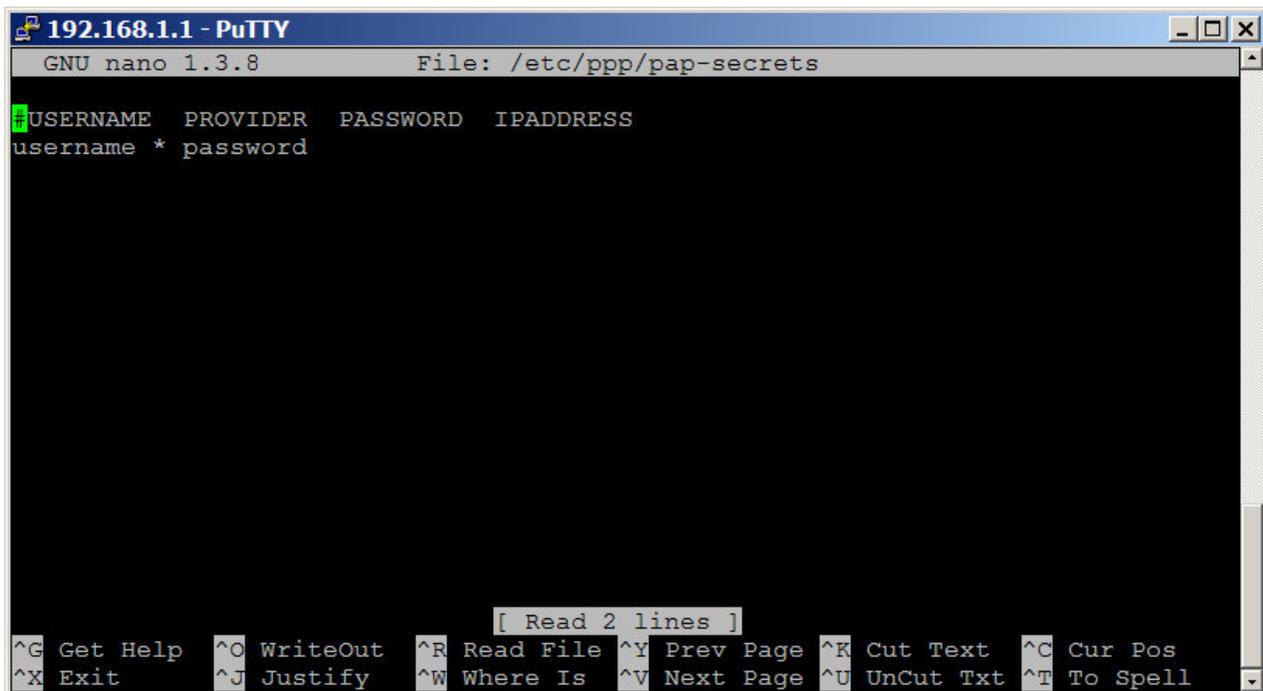
## STEP 9 Add your username and password

We have to add your bigpond username and password to the R100 so it will be able to log onto bigpond.

The files to edit are: `/etc/ppp/pap-secrets` and `/etc/ppp/peers/bigpond`

I have added a friendly editor called nano to make this easy. From the openwrt command prompt type:

```
nano /etc/ppp/papsecrets
```



```
192.168.1.1 - PuTTY
GNU nano 1.3.8 File: /etc/ppp/pap-secrets
#USERNAME PROVIDER PASSWORD IPADDRESS
username * password

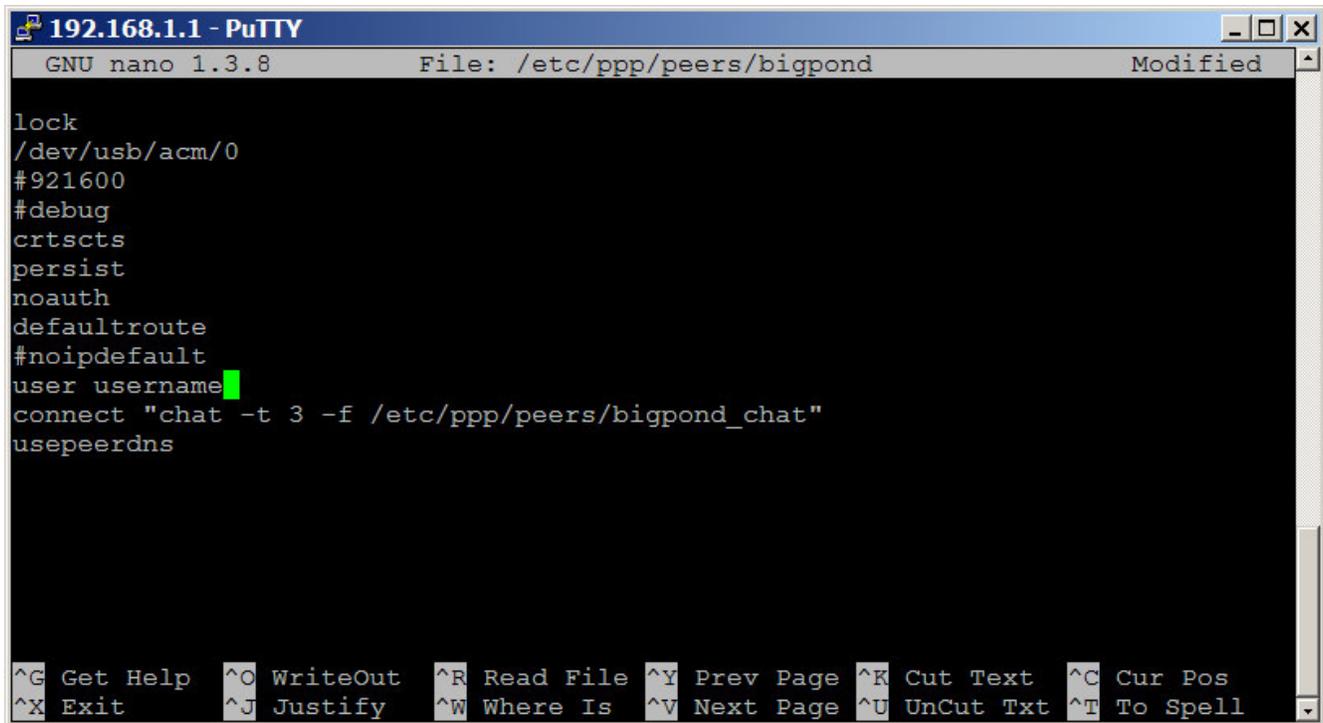
[ Read 2 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Txt ^T To Spell
```

nano works like a simple text editor. Arrow keys to move around, backspace deletes. Change only the lower case “username” and “password” to your actual username and password (ie the 2<sup>nd</sup> line). Take care not to delete other spaces and the asterix. The format of this file is important. Control X to exit. You do want to save (“y”).

Next file:

```
nano /etc/ppp/peers/bigpond
```

The only variable you are interested in is “user”. Replace “username” with your username from above. Save. This is all the configuration for bigpond ppp that you need to do.



```
192.168.1.1 - PuTTY
GNU nano 1.3.8 File: /etc/ppp/peers/bigpond Modified
lock
/dev/usb/acm/0
#921600
#debug
crtstcts
persist
noauth
defaultroute
#noipdefault
user username
connect "chat -t 3 -f /etc/ppp/peers/bigpond_chat"
usepeerdns

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Txt ^T To Spell
```

## **STEP 10. Configure NVRAM**

NVRAM is the flash memory set aside for storing system configuration variables. As you have just changed from the factory firmware, you will find many many nvram variables. Openwrt only needs a small snippet of these. I have written a little script that sorts these out for you automatically.... assuming we have the same default nvram variables – mine come from the ASUS wl500g 1.9.2.7.6b firmware (See DOS window 3). At the Openwrt command prompt type (case is important)

```
root@OpenWrt:~# nvram-ISDN
```

After 10 seconds or so, this will produce output like:

```
root@OpenWrt:~# nvram-ISDN
Before: size: 6028 bytes (26740 left)
After: size: 940 bytes (31828 left)
root@OpenWrt:~#
```

now type:

```
nvram show | sort
```

Which should produce a list like so:

```
root@OpenWrt:~# nvram show | sort
size: 688 bytes (32080 left)
boardnum=asusX
boardtype=bcm94710dev
boot_wait=on
```

```
clkfreq=125
dl_ram_addr=a0001000
et0macaddr=00:11:2F:4E:75:4E
et0mdcport=0
et0phyaddr=30
et1macaddr=00:11:2F:4E:75:4E
et1mdcport=1
et1phyaddr=0
hardware_version=WL500-02-02-01-00
lan_ifname=br0
lan_ifnames=eth0 eth2
lan_ipaddr=192.168.1.1
lan_netmask=255.255.255.0
lan_proto=static
os_date=Jun 29 2005
os_flash_addr=bfc40000
os_ram_addr=80001000
os_version=3.91.7.0
pmon_ver=PMON 3.11.19.0
scratch=a0180000
sdram_config=0x0000
sdram_init=0x0419
sdram_ncdl=0x00000000
sdram_refresh=0x8040
wan_device=ppp0
wan_ifname=ppp0
wan_proto=dhcp
watchdog=3000
wl0_ifname=eth2
wl0_infra=1
wl0_mode=ap
wl0_radio=1
wl0_ssid=OpenWrt
root@OpenWrt:~#
```

Make sure the list is close to this. The lan\_ and wan\_ variables are the important ones. These two

```
wan_device=ppp0 wan_ifname=ppp0
```

tell the R100 that its interface to the internet is ppp0 – ie the first ppp link (the ISDN modem!). All routing / forwarding / NATing is handled by those.

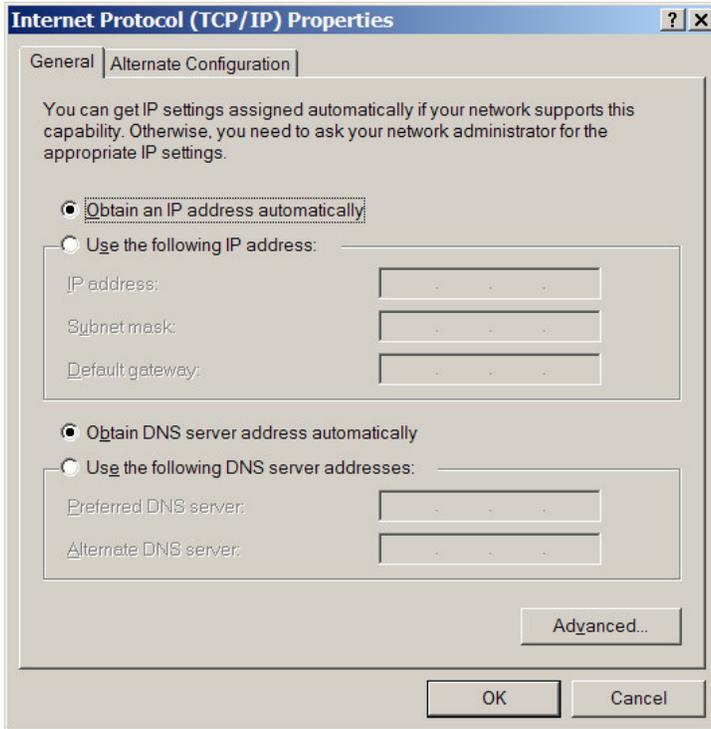
However, the nvram-ISDN script only changes the variables; it does not write (save) them to the flash memory. We have to commit them to flash, then reboot:

```
nvram commit
reboot
```

The router should again reboot.

## **STEP 11. Test DHCP.**

If you have wireless, you should now be able to connect to the OpenWrt wireless network and get an IP address on the 192.168.1.x subnet. Also change the configuration of the PC's Ethernet card back to DHCP:



and OK. You don't need to reboot the PC. The interface should now get an IP address from the R100. As below, my lan connection now gets 192.168.1.129, whilst the wireless gets 192.168.1.172. This is all good.

```
CA Command Prompt
D:\computer\R100>ipconfig

Windows IP Configuration

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  . : lan
    IP Address. . . . . : 192.168.1.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

Ethernet adapter Wireless Network Connection:

    Connection-specific DNS Suffix  . : lan
    IP Address. . . . . : 192.168.1.172
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.1.1

D:\computer\R100>
```

**STEP 12. Find the modem.**

Plug in the NT1 Plus II modem to USB port of the R100. At the R100 command prompt type

```
dmesg
```

The output should look something like

```
hub.c: new USB device 01:03.0-1, assigned address 2
ttyACM0: USB ACM device
```

This means the modem is recognised by the acm driver. Its address is actually `/dev/usb/acm/0`

**STEP 13. Connect to Bigpond.**

Dialing up should happen automatically. The pppd daemon (that handles this) might have exited for a timeout if it tries several times and it not successful (ie for eg if you don't have a modem connected). To force it to restart type at the R100 command prompt:

```
killall pppd
```

then

```
logread
```

You should be seeing the lights on the NT1Plus II change to red. If they flicker back and forth, then your username or password are incorrect. Logread should produce an output like this:

```
Jan 1 11:06:20 (none) kern.info init: ^MStarting pid 531, console /dev/tts/0: '/usr/sbin/pppd'  
Jan 1 11:06:20 (none) kern.notice pppd[531]: pppd 2.4.3 started by root, uid 0  
Jan 1 11:06:23 (none) kern.info pppd[531]: Serial connection established.  
Jan 1 11:06:23 (none) kern.info pppd[531]: Using interface ppp0  
Jan 1 11:06:23 (none) kern.notice pppd[531]: Connect: ppp0 <--> /dev/usb/acm/0  
Jan 1 11:06:26 (none) kern.notice pppd[531]: PAP authentication succeeded  
Jan 1 11:06:26 (none) kern.notice pppd[531]: PAP authentication succeeded  
Jan 1 11:06:26 (none) kern.notice pppd[531]: local IP address 144.134.217.103  
Jan 1 11:06:26 (none) kern.notice pppd[531]: remote IP address 139.134.58.233  
Jan 1 11:06:26 (none) kern.notice pppd[531]: primary DNS address 203.49.70.20  
Jan 1 11:06:26 (none) kern.notice pppd[531]: secondary DNS address 139.134.2.190  
Jan 1 11:06:26 (none) kern.info dnsmasq[481]: reading /tmp/resolv.conf  
Jan 1 11:06:26 (none) kern.info dnsmasq[481]: using nameserver 139.134.2.190#53  
Jan 1 11:06:26 (none) kern.info dnsmasq[481]: using nameserver 203.49.70.20#53  
Jan 1 11:06:26 (none) kern.info dnsmasq[481]: using local addresses only for domain lan
```

You are now up and running.

## **STEP 14 Configure Wireless**

I don't recommend to have an open wireless network. I recommend you configure a WPA network via the web interface. Point a browser at 192.168.1.1. Choose "Network". This will ask for user name / password. They are of course root / admin.

In wireless configuration select WPA(PSK) as the encryption setting, choose WPA1, RC4(TKIP) and set a password. Note that the password must be 8 or longer characters or it wont work. Save changes, apply changes. You should now be set.