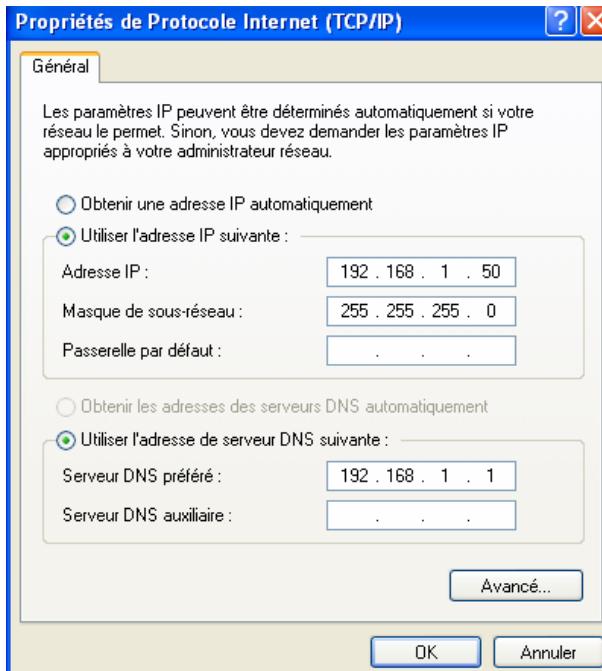Based on comments and experience, I have adjusted the manual. Forget about the web interface of the router, but follow these steps. It has proven that these settings work better for most (with thanks to DaNawq and other users for their help and comments):
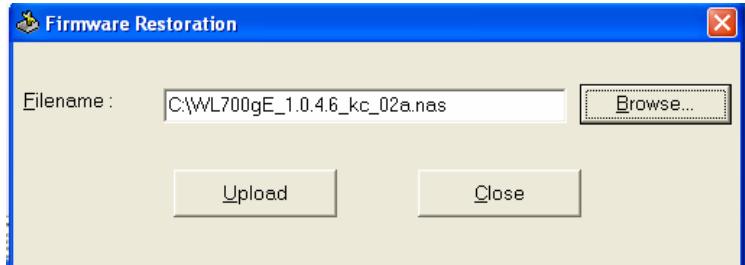

1. Before you do anything to your router, first give your desktop computer a static IP address like 192.168.1.50 (Subnet mask: 255.255.255.0). This is needed to be able to do the steps below. (From experience: Stop all the background programs like anti-virus of wirewall if the below steps don't work. Worse case, you need a clean Windows install), If you have more then 1 network card, it may also be usefull to disable the other cards. This way you will ensure the tftp server will attach it self to the network card with the above network address.

2. Now unplug the power cord of your router for about 10-20 seconds

3. Hold the orange EZSetup-button while plugging power on

4. Once the blue Ready LED is blinking unhold immediately EZSetup button.
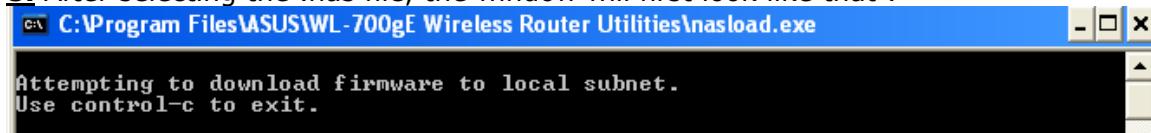
5. Wait 10 seconds (so the router can start up)

6. Start the Firmware-Restore tool on your desktop computer to upload a working firmware (do not start it before any of the earlier steps).
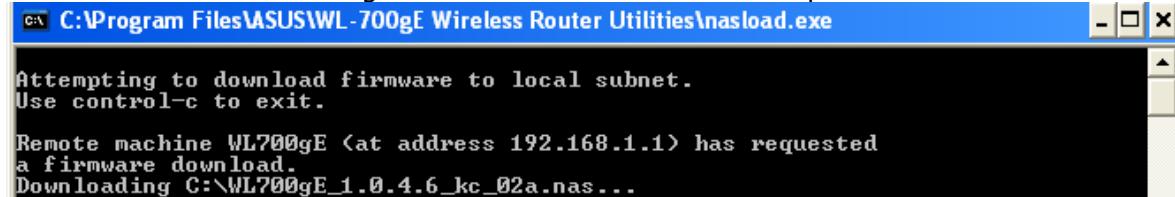
7. Press the Browse button to select a firmware file. E.g. "WL700gE_1.0.4.2.nas" and then press "Upload"

8. After selecting the .nas file, the window will first look like that :

It is now running a tftp server server on your machine. The tftp server is anounceing itself on the subnet mask.
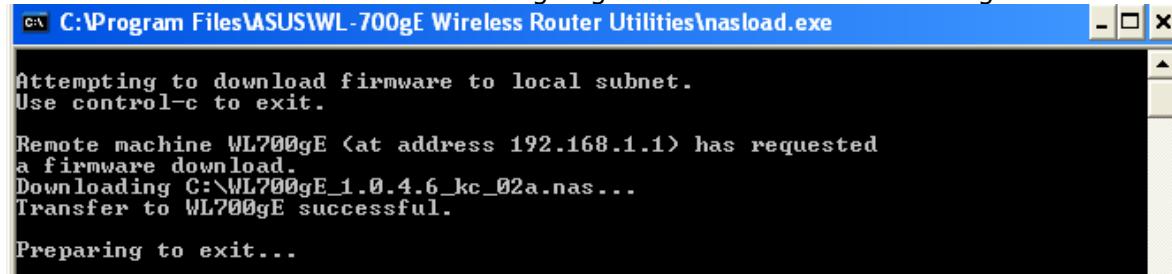Then the window will change once the router notices the tftp server:

You can now see the IP address of the server as well as it's name. You can also see that the Restoration software copied the .nas file to the root of your C-drive. (It doesn't delete it afterwarts, so you can do that yourself later)

And you see the that the router is downloading the firmware.
After about 1 min the window will change again and should be something like this one:


```
C:\Program Files\ASUS\WL-700gE Wireless Router Utilities\nasload.exe

Attempting to download firmware to local subnet.
Use control-c to exit.

Remote machine WL700gE (at address 192.168.1.1) has requested
a firmware download.
Downloading C:\WL700gE_1.0.4.6_kc_02a.nas...
Transfer to WL700gE successful.

Preparing to exit...
```
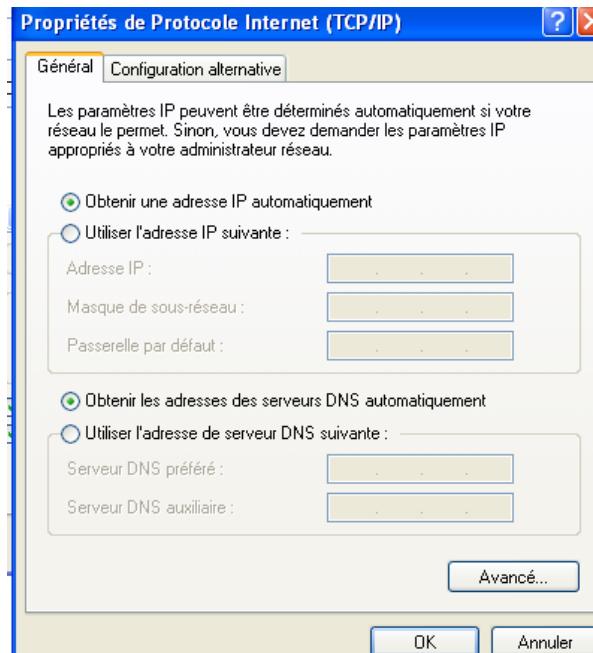
As you can see the transfer is successfull and the program will shutdown on your PC.

9. Do NOT turn off the router now, but wait for 1 or 2 minutes and you will notice that your router is still working even if in previous step the window automatically closed. The blue IDE LED will blink showing that your manchine is still processing some stuff and after a while the blue Ready LED and also the blue AIR LED will shine.
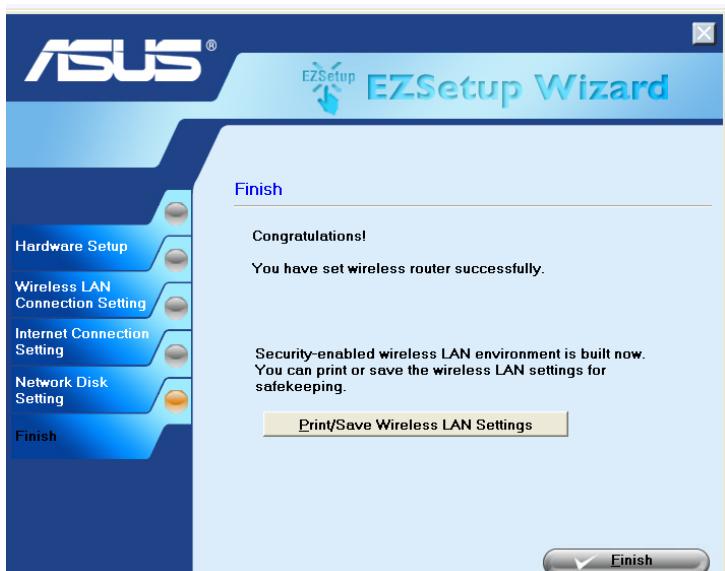
10. Do NOT !!!!! NOT Press the Reset button !!!!! If you would do this, the router will become useless (untill you repeat step 1 again)

11. Wait for the router to boot. After about a minute or so, the "Ready" light will come on.

12. On your PC, change back the static IP number to DHCP setttings.

13. Now you can press the EZSetup-button for a few seconds until the start button starts blinking to configure the router with the EZ Setup Wizard software.
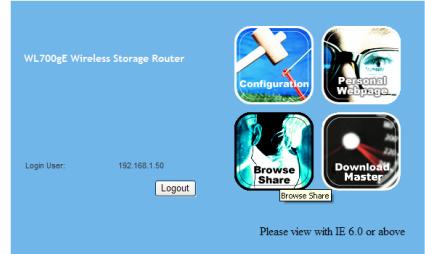
To customize the router startup, an rc.local file ***MUST*** be put onto the router. From the factory default state, with no telnet access, adding rc.local takes two steps. The first step allows the script to be executed from a location where it's easy to drop it onto the router. Unfortunately, this location is rather insecure. The second step secures the script from the outside world by relocating rc.local to an internal location not accessible via standard file sharing.

1. Grab the starter rc.local script. The script must have proper unix-style line terminators to execute on the router so don't edit it with braindead windoze editors like notepad.

Example rc.local files

1. This rc.local script starts an unprotected telnet daemon on the local interface, creates a functional fstab, and sets up the environment for installing and executing optware packages.
2. In addition to #1, this rc.local script spins down the internal HDD. You'll have to rename this file to rc.local and create a /shares/MYVOLUME1/opt/etc directory in order for the HDD spin-down to work properly (cd /shares/MYVOLUME1; mkdir -p opt/etc).
3. In addition to #2, this rc.local script does a lot more (it's intended for **demonstration only**! You certainly won't want to use it as-is.). You'll have to rename this file to rc.local and it also requires /shares/MYVOLUME1/opt/etc being present.

---

2. Copy rc.local into /shares/MYVOLUME1/MYSHARE1/web using one of the many file transfer methods. I've tested the ftp, cifs (samba), and nfs interfaces.
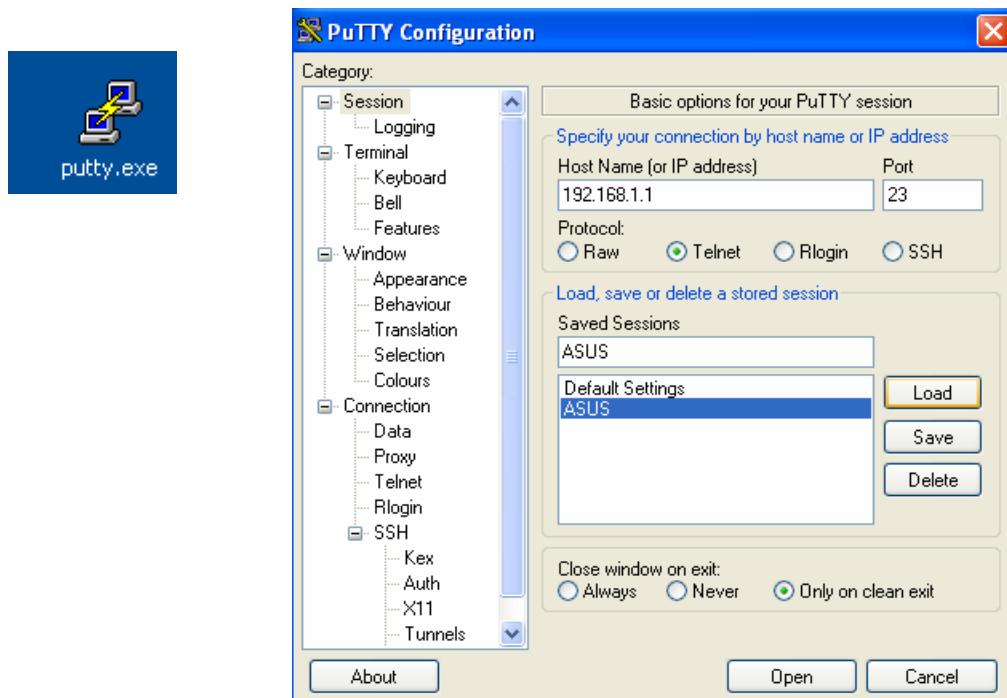

Contents of Directory /shares/MYSHARE1/web

Name	Size	Modification Time
..		
hint	directory	Sat Jan 20 20:33:55 2007
Layout_English	directory	Sat Jan 20 20:33:55 2007
Music	32	Sat Jan 20 20:34:01 2007
Layout	14	Sat Jan 20 20:34:05 2007
Layout_Japan	directory	Sat Jan 20 20:33:58 2007
MessageBoard	directory	Sat Jan 20 17:08:09 2007
PhotoAlbum	directory	Sat Jan 20 17:08:09 2007
index.html	442	Sat Jan 20 20:33:55 2007
Personal	directory	Sat Jan 20 17:08:09 2007
shares	directory	Sat Jan 20 17:08:09 2007
Layout_Taiwan	directory	Sat Jan 20 20:33:57 2007
psp.html	383	Sat Jan 20 20:33:55 2007
rc.local	1065	Sat Jan 20 21:53:55 2007

File Uploading

File to Upload into this Directory: ings\Admin\Bureau\rc.local [Parcourir...]

3. Reboot your router. You now have unsecured telnet access from the local LAN interface.

4. Telnet to your router. Move rc.local from /shares/MYVOLUME1/MYSHARE1/web to /shares/MYVOLUME1, and change its permissions to 755. (cd /shares/MYVOLUME1; mv MYSHARE1/web/rc.local .; chmod 755 rc.local).

5. Step #4 helps close the security hole opened by step #2. The wide-open telnet access is still a problem, but at least it's constrained to the internal LAN interface. This hole is easily closed after installing some optware packages.

For those of you who installed my previous firmware image, please note rc.local has moved from /shares/MYVOLUME1/MYSHARE1 to /shares/MYVOLUME1. If you don't move your existing rc.local to the new location prior to reflashing, you won't be able to get back into your router without starting from step #1!

1. Telnet to your router and login as root.

2. Create and manually mount /shares/MYVOLUME1/opt on /opt (rc.local will do this automatically once /shares/MYVOLUME1/opt is created)
   - cd /shares/MYVOLUME1
   - mkdir -p opt/etc
   - mount -o bind /shares/MYVOLUME1/opt /opt

```
#
# cd /shares/MYVOLUME1
#
# mkdir -p opt/etc
#
# mount -o bind /shares/MYVOLUME1/opt /opt
#
#
```

3. Manually install uClibc, busybox, wget, and ipkg from the optware repository:
   - wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/uclibc_0.9.28-1_mipsel.ipk
   - tar xvzf uclibc_0.9.28-1_mipsel.ipk; tar xvzf data.tar.gz

```
# wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/uclibc_0.9.28
-1_mipsel.ipk
#
# tar xvzf uclibc_0.9.28-1_mipsel.ipk
./debian-binary
./data.tar.gz
./control.tar.gz
#
# tar xvzf data.tar.gz
.
./opt
./opt/sbin
./opt/sbin/ldconfig
./opt/lib
./opt/lib/librt.so.0
```

   - wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/busybox-base_1.1.3-3_mipsel.ipk
   - tar xvzf busybox-base_1.1.3-3_mipsel.ipk; tar xvzf data.tar.gz

```
# wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/busybox-base_
1.1.3-3_mipsel.ipk
#
# tar xvzf busybox-base_1.1.3-3_mipsel.ipk
./debian-binary
./data.tar.gz
./control.tar.gz
#
# tar xvzf data.tar.gz
.
./opt
./opt/bin
./opt/bin/busybox
#
```

- wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/busybox-links_1.1.3-3_mipsel.ipk
- tar xvzf busybox-links_1.1.3-3_mipsel.ipk; tar xvzf data.tar.gz

```
# wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/busybox-links
_1.1.3-3_mipsel.ipk
#
# tar xvzf busybox-links_1.1.3-3_mipsel.ipk
./debian-binary
./data.tar.gz
./control.tar.gz
#
# tar xvzf data.tar.gz
```

- wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/wget_1.10.2-3_mipsel.ipk
- tar xvzf wget_1.10.2-3_mipsel.ipk; tar xvzf data.tar.gz

```
# wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/wget_1.10.2-3
_mipsel.ipk
#
# tar xvzf wget_1.10.2-3_mipsel.ipk
./debian-binary
./data.tar.gz
./control.tar.gz
#
# tar xvzf data.tar.gz
.
./opt
./opt/man
./opt/man/man1
```

- wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/ipkg_0.99-163-4_mipsel.ipk
- tar xvzf ipkg_0.99-163-4_mipsel.ipk; tar xvzf data.tar.gz

```
# wget http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable/ipkg_0.99-163
-4_mipsel.ipk
#
# tar xvzf ipkg_0.99-163-4_mipsel.ipk
./debian-binary
./data.tar.gz
./control.tar.gz
#
# tar xvzf data.tar.gz
.
./opt
./opt/bin
./opt/bin/ipkg
```

- rm -f *.ipk data.tar.gz debian-binary control.tar.gz

```
#
# rm -f *.ipk data.tar.gz debian-binary control.tar.gz
#
#
```

4. Add /opt/bin to the beginning your path (to override Asus binaries of the same name)
   - export PATH=/opt/bin:${PATH}
   - export LD_LIBRARY_PATH=/opt/lib:${LD_LIBRARY_PATH}

```
#
# export PATH=/opt/bin:${PATH}
#
# export LD_LIBRARY_PATH=/opt/lib:${LD_LIBRARY_PATH}
#
```

5. Add the following line to /opt/etc/ipkg.conf with the newly installed busybox version of vi: "src optware http://ipkg.nslu2-linux.org/feeds/optware/oleg/cross/stable". Here's a copy of mine.

6. Run "ipkg update"

Sorry but after this step i don't know how to do….. so I can't give printscreen !

7. For completeness, re-run "ipkg install" for uclibc, busybox, ipkg, and wget to fully install them. You may have to use the -force-overwrite flag.
8. You can now add packages as necessary and start them via rc.local

Step #4, Closing the telnet hole

With the default config, root access via telnet will be wide open with no password protection. Any user accounts added by the web interface will be password protected with /bin/sh as their default shell. Either manually edit /etc/passwd and use an existing user account password as the root password, or install dropbear and disable telnet altogether (which is what I did).

Step #5, Making a better system

1. Install the bash package for a better shell
2. Install the sudo package so you don't have to login as root to perform administrative functions (you can't login as root via ssh anyway)
3. Give yourself supreme permissions in /opt/etc/sudoers
4. Add /opt/bin/bash to /etc/shells (echo "/opt/bin/bash" >> /etc/shells)
5. Add a non-root user account via the web interface
6. Give yourself a home directory (mkdir -p /opt/home/<username>; chown <username> /opt/home/<username>
7. Modify /etc/passed to make /opt/bin/bash your login shell and /opt/home/<username> your home directory (vi /etc/passwd). See here for a non-functional sample.
8. Optionally symlink your user account directory into /shares/MYVOLUME1/MYSHARE1 if you want it accessable via cifs (Samba). This trick won't work for ftp, nfs or the web interface. You'll have to make your home directory in /shares/MYVOLUME1/MYSHARE1 if you want full access from all methods. (cd /shares/MYVOLUME1/MYSHARE1; ln -s /opt/home/<username>)
9. Within your .profile, add /opt/bin and /opt/lib to your PATH and LD_LIBRARY_PATH environment variables
10. Install dropbear and start it (manually at first, then via rc.local), see the last rc.local file below.
11. Disable telnet by removing it from rc.local to completely eliminate any chance of root logins
12. Install ncurses, ncursesw, and termcap packages for better terminal support. You may need to use the -force-overwrite option when installing ncurses.
13. Install man and manpages packages