

Tutorial 1: GNU Radio Installation Guide - Step by Step

Dawei Shen*

May 19, 2005

Abstract

The tutorials for GNU Radio start from here. The installation procedure is introduced in detail. It is shown that many packages are needed by GNU Radio, which makes the installation bothersome. To avoid trouble, Fedora and Mandrake are recommended. For first time users, it's better to install Linux completely and follow step 1 to step 7 strictly.

The best and easiest way to get GNU Radio software is to build and install it directly from CVS, which is described in Step 8. However, GNU Radio requires **many** other packages pre-installed in your system. Missing any one of them will lead you into trouble. So I suggest for first time installation, it is better to go through 1 until 7 step by step. Later, whenever you want to check new versions of GNU Radio appearing on CVS, you could go to step 8 directly. GNU Radio is designed and tested well in Linux operating system. **Fedora** and **Mandrake** seem to work best at this point. Make sure when you install Linux, the development tools and the x-window tools are included, which can make your life much easier. Complete installation of Linux is a good option for beginners.

1 Downloading necessary packages or tarballs

For our *USRP* users, we need the following modules given in Table 1.

*The author is affiliated with Networking Communications and Information Processing (NCIP) Laboratory, Department of Electrical Engineering, University of Notre Dame. Welcome to send me your comments or inquiries. Email: dshen@nd.edu

Table 1: GNU Radio Modules

gnuradio-core (*)	the 2.x core library
gnuradio-examples	examples for the 2.x tree
gr-audio-oss (*)	soundcard support using OSS
gr-audio-alsa (*)	soundcard support using ALSA
gr-usrp (*)	glue that connects usrp into GNU Radio framework
gr-wxgui (*)	wxPython based GUI tools including FFT and o'scope
gr-howto-write-a-block	examples, Makefiles and article source
gr-gsm-fr-vocoder	GSM 06.10 13kbit/sec vocoder
usrp (*)	USRP board support

The modules marked with ‘*’ are necessary components. These modules can be downloaded directly from the GNU Radio’s web page. However, the best way to get the newest version of these modules (except *usrp*) is to checkout and download them from CVS using the following commands:

```
$export CVS_RSH="ssh"
$cvcs -z3 -d:ext:anoncvcs@savannah.gnu.org:/cvsroot/gnuradio co -P <modulename>
```

The *USRP* CVS tree is hosted on SourceForge. Use these commands to check the *usrp* module out:

```
$cvcs -d:pserver:anonymous@cvs.sourceforge.net:/cvsroot/opensdr login
$cvcs -z3 -d:pserver:anonymous@cvs.sourceforge.net:/cvsroot/opensdr co -P usrp
```

Then a directory with the module name will appear in your current folder. Every time you should run `./bootstrap` first to generate the configuration and make files.

2 Installation of gnuradio-core

This is the most important module in GNU Radio, which requires many packages pre-installed in your system. To make our life easier, when we install the GNU/Linux in our computer (Fedora 3 for me), we should make sure the **development tools** and **X-window development tools** are checked, so that we can avoid a lot of trouble. It has been determined that GNU Radio exercises bugs in certain versions of g++ 3.3.x on the x86 platform. If you are using g++ 3.3 and make check fails, please either upgrade to 3.4 or downgrade to 3.2. Both are known to work.

Three packages are necessary and often missing:

A. *FFTW*

Webpage: <http://www.fftw.org/>

Download page: <http://www.fftw.org/download.html>

IMPORTANT!!! When building *FFTW*, you MUST use the `--enable-single` and `--enable-shared` configure options. You should also use either the `--enable-3dnow` or `--enable-sse` options if you’re on an Athlon or Pentium respectively. Note that sometimes you may meet the ‘segmentation fault’ error when you try to run a python code. You can remove the `--enable-sse` option to avoid this problem.

B. *cppunit*

Webpage: <http://cppunit.sourceforge.net>

Download page: https://sourceforge.net/project/showfiles.php?group_id=11795

C. *SWIG*

Webpage: <http://www.swig.org>

Download page: <http://sourceforge.net/projects/swig/>

Two Python modules are necessary and very important in GNU Radio, *Numeric* and *Numarray*, which are used in Python for numerical computation purpose. They are available at:

<http://sourceforge.net/projects/numpy>

Download the newest version of *Numarray* and *Numpy*. Extract the tarball then you can find the `setup.py` file. Use the following command to install

```
$python setup.py install
```

At this point, I guess you should be able to install *gnuradio-core* without losing any packages. In the folder *gnuradio-core* downloaded from CVS, run the following commands

```
$ ./bootstrap
$ ./configure --enable-maintainer-mode
$ make
$ make check
$ make install
```

Now you have already got the key component of GNU Radio. Note that you may not be able to pass the ‘make check’ if you didn’t install *Numeric* and *Numarray* modules.

3 Installation of gr-audio-oss, gr-audio-alsa

Nothing particular is worth mentioning here, they are two modules providing you the access to your sound card. Remember to run `./bootstrap` first to generate the configuration and make files.

4 Installation of gr-wxgui

gr-wxgui provides the graphical FFT and o’scope in GNU Radio and is based on *wxPython*. So certainly we should install *wxPython* first.

Webpage: <http://www.wxPython.org/>

Download page: https://sourceforge.net/project/showfiles.php?group_id=10718

There are two types of *wxPython*, which are ansi based and unicode based. Either one of them works. Pay attention to the suffix and choose the correct rpms for your Python (2.3 or 2.4). For example, in my computer, I choose the following three rpms.

```
wxPython2.5-gtk2-unicode-2.5.4.1-fc2_py2.3.i386.rpm
wxPython-common-gtk2-unicode-2.5.4.1-fc2_py2.3.i386.rpm
wxPython2.5-devel-gtk2-unicode-2.5.4.1-fc2_py2.3.i386.rpm
```

Download them into one folder and simply run

```
$ rpm -ivh *.rpm
```

Now *wxPython* has been installed and you can install *gr-wxgui* without any problem.

5 Installation of usrp and gr-usrp

The *usrp* module provides necessary drivers, fpga controls, firmware interfaces for GNU Radio’s use. It must be installed before *gr-usrp*. We’ll need the *SDCC* free C compiler to build the firmware. *SDCC* can be found at

Webpage: <http://sdcc.sourceforge.net/>

Download page: http://sourceforge.net/project/showfiles.php?group_id=599

The version 2.4 is necessary to avoid some problems. Now you can install *usrp* and *gr-usrp* in order.

6 Installation of gnuradio-examples, gr-howto-write-a-block and gr-gsm-fr-vocoder

Actually these modules are not parts of the GNU Radio software. They just provide some examples, applications or technical articles. They can be installed easily. Note that it's not necessary to 'install' *gnuradio-examples*, you can find many good examples in the `/python` folder.

7 Post-installation tasks

After installation, all the components mentioned above are installed in

```
/usr/local/lib/python2.3/site-packages
```

If you are using python 2.4, then the path should be

```
/usr/local/lib/python2.4/site-packages
```

which is not included in Python's working path by default. To let the Python find these modules and for convenience, you'll need to set the `PYTHONPATH` environment variable as follows:

```
$ export PYTHONPATH=/usr/local/lib/python2.3/site-packages
```

You may want to add this to your `~/.bash_profile`.

Two important documents are located at

```
/usr/local/share/doc/gnuradio-core-x.xcvs/html/index.html
/usr/local/share/doc/usrp-x.xcvs/html/index.html
```

They are the documentations for *gnuradio* and *usrp*. You may like to bookmark them in your web browser.

8 How to check the updates and install the updated versions of GNU Radio software from CVS?

Note that the steps mentioned above are only for first-time installation, because typically quite a lot of packages are missing at this point. Once you have installed GNU Radio successfully, you have a much easier way to check the updates and install the updated versions of GNU Radio from CVS. Here's the step-by-step version.

We checkout *gr-build* by hand, then use it to bootstrap everything else.

```
$ export CVS_RSH="ssh"
$ cvs -z3 -d:ext:anoncvs@savannah.gnu.org:/cvsroot/gnuradio co -P gr-build
```

You now have a directory called *gr-build*. Change directory into it:

```
$cd gr-build
```

This checkout command will checkout everything except for that related to the Measurement Computing PCI card.

```
$/checkout -x mc4020
```

You'll now have a bunch of directories...

If you haven't already, you'll want to set up */etc/sudoers*. The *buildit* script below does the bulk of the work running as you, then uses '**sudo make install**' to install the stuff into */usr/local/...*

First, add yourself to group **wheel** by editing */etc/group*. Find the line that starts with **wheel:x:...** and add your log name to the end of it.

Then make these changes to */etc/sudoers*:

```
# Defaults specification
# Only ask for password every 10 minutes.
Defaults timestamp_timeout=10
# Uncomment to allow people in group wheel to run all commands
%wheel ALL=(ALL) ALL
```

Now, assuming you've got all the dependencies fulfilled, this will bootstrap, configure, make, make check, and make install everything in the proper order:

```
$sudo -v # give sudo the password now, so you can walk away while it builds
```

Now build everything that you checked out

```
$/for-all-dirs ../buildit 2>&1 | tee make.log
```

Assuming the build completed successfully, you're all set.

Later when you want to see if there are updates to CVS head, but don't want to apply them to your tree use this:

```
$/for-all-dirs cvs -nqP up
```

If you like what you saw from the previous command, this will get the updates and merge them into your tree:

```
$/for-all-dirs cvs -qP up
```

Then you should rebuild everything:

```
$/for-all-dirs ../buildit
```

References

- [1] GNU Radio on-line document: **How to Build from CVS**,
<http://comsec.com/wiki?HowtoBuildFromCVS>