Getting Started V2.8.0~pre1-ja~joints-axes13~766c022, 2016-05-11

Contents

1 About LinuxCNC **1**

1.1 The Software . . . 1

1.2 The Operating System . . . 1

1.3 Getting Help . . . 2

1.3.1 IRC . . . 2

1.3.2 Mailing List . . . 2

1.3.3 LinuxCNC Wiki . . . 2

1.3.4 Bug Reports . . . 2

2 System Requirements **3**

2.1 Minimum Requirements . . . 3

2.2 Problematic Hardware . . . 3

2.2.1 Laptops . . . 3

2.2.2 Video Cards . . . 3

3 Getting LinuxCNC **4**

3.1 Download the image . . . 4

3.1.1 Normal Download . . . 4

3.1.2 Download using zsync . . . 4

3.1.3 Verify the image . . . 5

3.2 Write the image to a bootable device . . . 5

3.3 Testing LinuxCNC . . . 6

3.4 Installing LinuxCNC . . . 6

3.5 Updates to LinuxCNC . . . 6

3.6 Install Problems . . . 6

3.7 Alternate Install Methods . . . 6

3.7.1 Installing on Debian Wheezy (with Preempt-RT kernel) . . . 7

3.7.2 Installing on Ubuntu Precise . . . 8

4 Updating LinuxCNC **9**
4.1 Upgrade to the new version . . . 9
4.1.1 Setting apt sources . . . 9
4.1.2 Upgrading to the new version . . . 11
4.1.2.1 Debian Wheezy and Ubuntu Lucid . . . 11
4.1.3 Ubuntu Precise . . . 11
4.2 Updating Configuration Files . . . 11
4.2.1 Distribution Configurations (updates for joints_axes) . . . 11
4.2.2 Automatic updates (update_ini script for joints_axes) . . . 11
4.2.3 Kinematics modules . . . 12
4.2.4 Lathe Configurations . . . 12
4.2.5 Consistent Joints/Axes specifications . . . 13
4.2.6 Home sequences . . . 14
4.2.7 Locking rotary indexer (updates for joints_axes) . . . 14
4.2.8 Stricter INI file syntax . . . 15
4.3 Hal Changes (updates for joints_axes) . . . 15
4.3.1 Wheel or MPG (manual pulse generator) jogging . . . 15
4.3.2 Ini Hal pins . . . 16
4.4 GUIs (updates for joints_axes) . . . 16
4.4.1 Halui . . . 16
4.4.1.1 TELEOP jogging (also called axis or world jogging) . . . 17
4.4.1.2 Joint jogging . . . 17
4.4.1.3 Aditional pin renames . . . 17
4.4.2 Axis gui . . . 17
4.4.2.1 Kinematics Support . . . 17
4.4.2.2 AUTO_TELEOP . . . 18
4.4.2.3 Home icons . . . 18
4.4.2.4 Limit icons . . . 18
4.4.2.5 Key bindings for a fourth axis . . . 18
4.4.3 tklinuxcnc . . . 18
4.4.3.1 Kinematics Support . . . 18
4.4.4 touchy . . . 19
4.4.5 gscreen . . . 19
4.4.6 gmoccapy . . . 19
4.4.7 linuxcnclcd . . . 19
4.4.8 linuxcncrsh . . . 19
4.5 Simulator configurations (updates for joints axes) . . . 19
4.5.1 Pre-joints_axes . . . 19
4.5.2 Post-joints_axes . . . 20
4.5.3 Notes . . . 20

5 Glossary **21**

6 Legal Section **26**

6.1 Copyright Terms 26

6.2 GNU Free Documentation License 26

7 Index **30**

The LinuxCNC Team

This handbook is a work in progress. If you are able to help with writing, editing, or graphic preparation please contact any member of the writing team or join and send an email to emc-users@lists.sourceforge.net.

Chapter 1

About LinuxCNC

1.1 The Software

- LinuxCNC (the Enhanced Machine Control) is a software system for computer control of machine tools such as milling machines and lathes, robots such as puma and scara and other computer controlled machines up to 9 axes.
- LinuxCNC is free software with open source code. Current versions of LinuxCNC are entirely licensed under the GNU General Public License and Lesser GNU General Public License (GPL and LGPL)
- LinuxCNC provides:
  - a graphical user interface (actually several interfaces to choose from)
  - an interpreter for *G-code* (the RS-274 machine tool programming language)
  - a realtime motion planning system with look-ahead
  - operation of low-level machine electronics such as sensors and motor drives
  - an easy to use *breadboard* layer for quickly creating a unique configuration for your machine
  - a software PLC programmable with ladder diagrams
  - easy installation with a Live-CD
- It does not provide drawing (CAD - Computer Aided Design) or G-code generation from the drawing (CAM - Computer Automated Manufacturing) functions.
- It can simultaneously move up to 9 axes and supports a variety of interfaces.
- The control can operate true servos (analog or PWM) with the feedback loop closed by the LinuxCNC software at the computer, or open loop with step-servos or stepper motors.
- Motion control features include: cutter radius and length compensation, path deviation limited to a specified tolerance, lathe threading, synchronized axis motion, adaptive feedrate, operator feed override, and constant velocity control.
- Support for non-Cartesian motion systems is provided via custom kinematics modules. Available architectures include hexapods (Stewart platforms and similar concepts) and systems with rotary joints to provide motion such as PUMA or SCARA robots.
- LinuxCNC runs on Linux using real time extensions.

1.2 The Operating System

LinuxCNC is available as ready-to-use packages for the Ubuntu and Debian distributions.

1.3 Getting Help

1.3.1 IRC

IRC stands for Internet Relay Chat. It is a live connection to other LinuxCNC users. The LinuxCNC IRC channel is #linuxcnc on freenode.

The simplest way to get on the IRC is to use the embedded java client on this page.

SOME IRC ETIQUETTE

- Ask specific questions... Avoid questions like "Can someone help me?".
- If you're really new to all this, think a bit about your question before typing it. Make sure you give enough information so someone can solve your question.
- Have some patience when waiting for an answer, sometimes it takes a while to formulate an answer or everyone might be busy working or something.
- Set up your IRC account with your unique name so people will know who you are. If you use the java client, use the same name every time you log in. This helps people remember who you are and if you have been on before many will remember the past discussions which saves time on both ends.

Sharing Files The most common way to share files on the IRC is to upload the file to one of the following or a similar service and paste the link:

- *For text* - http://pastebin.com/ , http://pastie.org/, https://gist.github.com/
- *For pictures* - http://imagebin.org/ , http://imgur.com/ , http://bayimg.com/
- *For files* - https://filedropper.com/ , http://filefactory.com/ , http://1fichier.com/

1.3.2 Mailing List

An Internet Mailing List is a way to put questions out for everyone on that list to see and answer at their convenience. You get better exposure to your questions on a mailing list than on the IRC but answers take longer. In a nutshell you e-mail a message to the list and either get daily digests or individual replies back depending on how you set up your account.

You can subscribe to the emc-users mailing list at: https://lists.sourceforge.net/lists/listinfo/emc-users

1.3.3 LinuxCNC Wiki

A Wiki site is a user maintained web site that anyone can add to or edit.

The user maintained LinuxCNC Wiki site contains a wealth of information and tips at:

http://wiki.linuxcnc.org

1.3.4 Bug Reports

Report bugs to the LinuxCNC github bug tracker.

Chapter 2

System Requirements

2.1 Minimum Requirements

The minimum system to run LinuxCNC and Ubuntu may vary depending on the exact usage. Stepper systems in general require faster threads to generate step pulses than servo systems. Using the Live-CD you can test the software before committing a computer. Keep in mind that the Latency Test numbers are more important than the processor speed for software step generation. More information on the Latency Test is here.

Additional information is on the LinuxCNC Wiki site:

Wiki.LinuxCNC.org, Hardware_Requirements

LinuxCNC and Ubuntu should run reasonably well on a computer with the following minimum hardware specification. These numbers are not the absolute minimum but will give reasonable performance for most stepper systems.

- 700 MHz x86 processor (1.2 GHz x86 processor recommended)
- 384 MB of RAM (512 MB up to 1 GB recommended)
- 8 GB hard disk
- Graphics card capable of at least 1024x768 resolution, which is not using the NVidia or ATI fglrx proprietary drivers, and which is not an onboard video chipset that shares main memory with the CPU
- A network or Internet connection (not strictly needed, but very useful for updates and for communicating with the LinuxCNC community)

Minimum hardware requirements change as Ubuntu evolves so check the Ubuntu web site for details on the LiveCD you're using. Older hardware may benefit from selecting an older version of the LiveCD when available.

2.2 Problematic Hardware

2.2.1 Laptops

Laptops are not generally suited to real time software step generation. Again a Latency Test run for an extended time will give you the info you need to determine suitability.

2.2.2 Video Cards

If your installation pops up with 800 x 600 screen resolution then most likely Ubuntu does not recognize your video card or monitor. Onboard video many times causes bad real time performance.

Chapter 3

Getting LinuxCNC

This section describes the recommended way to download and make a fresh install of LinuxCNC. There are also Alternate Install Methods for the adventurous. If you have an existing install that you want to upgrade, go to the Updating LinuxCNC section instead.

Fresh installs of LinuxCNC are most easily created using the Live/Install Image. This is a hybrid ISO filesystem image that can be written to a USB storage device or a DVD and used to boot a computer. At boot time you will be given a choice of booting the "Live" system (to run LinuxCNC without making any permanent changes to your computer) or booting the Installer (to install LinuxCNC and its operating system onto your computer's hard drive).

The outline of the process looks like this:

1. Download the Live/Install Image.
2. Write the image to a USB storage device or DVD.
3. Boot the Live system to test out LinuxCNC.
4. Boot the Installer to install LinuxCNC.

3.1 Download the image

This section describes some methods for downloading the Live/Install Image.

3.1.1 Normal Download

Download the Live/Install CD by clicking here:

http://www.linuxcnc.org/linuxcnc-2.7-wheezy.iso

3.1.2 Download using zsync

zsync is a download application that efficiently resumes interrupted downloads and efficiently transfers large files with small modifications (if you have an older local copy). Use zsync if you have trouble downloading the image using the Normal Download method.

ZSYNC IN LINUX

1. Install zsync using Synaptic or, by running the following in a terminal

```
sudo apt-get install zsync
```

2. Then run this command to download the iso to your computer

```
zsync http://www.linuxcnc.org/linuxcnc-2.7-wheezy.iso.zsync
```

zsync in Windows There is a Windows port of zsync. It works as a console application. It can be downloaded from:

https://www.assembla.com/spaces/zsync-windows/documents

3.1.3 Verify the image

(This step is unnecessary if you used zsync)

1. After downloading, verify the checksum of the image to ensure integrity.

```
md5sum linuxcnc-2.7-wheezy.iso
or
sha256sum linuxcnc-2.7-wheezy.iso
```

2. Then compare to these checksums

```
md5sum: 978ca074c51194e72f93e8c8d7110cfa
sha256sum: a3c29850cbc44da7b1ecdbe584a915f158c0b84428acfbcf3271df85c24e34aa
```

Verify md5sum on Windows or Mac Windows and Mac OS X do not come with an md5sum program, but there are alternatives. More information can be found at: How To MD5SUM

3.2 Write the image to a bootable device

The LinuxCNC Live/Install Image is a hybrid ISO image which can be written directly to a USB storage device (flash drive) or a DVD and used to boot a computer. The image is too large to fit on a CD.

1. Writing the image to a USB storage device in Linux
2. Connect a USB storage device (for example a flash drive or thumb drive type device).
3. Determine the device file corresponding to the USB flash drive. This information can be found in the output of `dmesg` after connecting the device. `/proc/partitions` may also be helpful.
4. Use the `dd` command to write the image to your USB storage device. For example, if your storage device showed up as `/dev/sde`, then use this command:

```
dd if=linuxcnc-2.7-wheezy.iso of=/dev/sde
```

WRITING THE IMAGE TO A DVD IN LINUX

1. Insert a blank DVD into your burner. A *CD/DVD Creator* or *Choose Disc Type* window will pop up. Close this, as we will not be using it.
2. Browse to the downloaded image in the file browser.
3. Right click on the ISO image file and choose Write to Disc.
4. Select the write speed. It is recommended that you write at the lowest possible speed.
5. Start the burning process.
6. If a *choose a file name for the disc image* window pops up, just pick OK.

Writing the image to a DVD in Windows

1. Download and install Infra Recorder, a free and open source image burning program: http://infrarecorder.org/
2. Insert a blank CD in the drive and select Do nothing or Cancel if an auto-run dialog pops up.
3. Open Infra Recorder, and select the *Actions* menu, then *Burn image*.

3.3 Testing LinuxCNC

With the USB storage device plugged in or the DVD in the DVD drive, the shut down the computer then turn the computer back on. This will boot the computer from the Live/Install Image and choose the Live boot option. Once the computer has booted up you can try out LinuxCNC without installing it. You can not create custom configurations or modify most system settings like screen resolution unless you install LinuxCNC.

To try out LinuxCNC from the Applications/CNC menu pick LinuxCNC. Then select a sim configuration to try out.

To see if your computer is suitable for software step pulse generation run the Latency Test as shown here.

3.4 Installing LinuxCNC

To install LinuxCNC from the LiveCD select *Install (Graphical)* at bootup.

3.5 Updates to LinuxCNC

With the normal install the Update Manager will notify you of updates to LinuxCNC when you go on line and allow you to easily upgrade with no Linux knowledge needed. It is OK to upgrade everything except the operating system when asked to.

Warning
Do not upgrade the operating system if prompted to do so.

3.6 Install Problems

In rare cases you might have to reset the BIOS to default settings if during the Live CD install it cannot recognize the hard drive during the boot up.

3.7 Alternate Install Methods

The easiest, preferred way to install LinuxCNC is to use the Live/Install Image as described above. That method is as simple and reliable as we can make it, and is suitable for novice users and experienced users alike.

In addition, for experienced users who are familiar with Debian system administration (finding install images, manipulating apt sources, changing kernel flavors, etc), new installs are supported on following platforms:

Distribution	Architecture	kernel	Typical use
Debian Jessie	amd64 & i386	Stock	simulation only
Debian Wheezy	i386	RTAI	machine control & simulation

Distribution	Architecture	kernel	Typical use
Debian Wheezy	amd64 & i386	Preempt-RT	machine control & simulation
Debian Wheezy	amd64 & i386	Stock	simulation only
Ubuntu Precise	i386	RTAI	machine control & simulation
Ubuntu Precise	amd64 & i386	Stock	simulation only
Ubuntu Lucid	i386	RTAI	machine control & simulation
Ubuntu Lucid	amd64 & i386	Stock	simulation only

The RTAI kernels are available for download from the linuxcnc.org debian archive. The apt source is:

- Debian Wheezy: `deb http://linuxcnc.org wheezy base`
- Ubuntu Precise: `deb http://linuxcnc.org precise base`
- Ubuntu Lucid: `deb http://linuxcnc.org lucid base`

The Preempt-RT kernels are available for Debian Wheezy from the regular debian.org archive. The packages are called `linux-image-rt-amd64` and `linux-image-rt-686-pae`.

3.7.1 Installing on Debian Wheezy (with Preempt-RT kernel)

1. Install Debian Wheezy (Debian version 7), either i386 or amd64. You can download the installer here: https://www.debian.org/-releases/. One version that is tested is the net install *debian-7.9.0-i386-netinst.iso*. Be careful and don't download Debian 8.
2. After burning the iso and booting up if you don't want Gnome desktop select *Advanced Options* > *Alternative desktop environments* and pick the one you like. Then select *Install* or *Graphical Install*.

> **Warning**
> Do not enter a root password, if you do sudo is disabled and you won't be able to complete the following steps.

3. Run the following in a terminal to bring the machine up to date with the latest packages.

```
sudo apt-get update
sudo apt-get dist-upgrade
```

4. Install the Preempt-RT kernel and modules

```
sudo apt-get install linux-image-rt-amd64
or
sudo apt-get install linux-image-rt-686-pae
```

5. Reboot, and make sure you boot into the Preempt-RT kernel. When you log in, verify that the following includes `PREEMPT` and `RT`.

```
uname -v
```

6. Add the LinuxCNC Archive Signing Key to your apt keyring by running

```
sudo apt-key adv --keyserver hkp://keys.gnupg.net --recv-key 3cb9fd148f374fef
```

7. Add a new apt source that looks like this:

```
sudo add-apt-repository "deb http://linuxcnc.org/ wheezy base 2.7-uspace"
or
sudo add-apt-repository "deb http://linuxcnc.org/ jessie base 2.7-uspace"
```

8. Update the package list from linuxcnc.org

```
sudo apt-get update
```

9. Install uspace

```
sudo apt-get install linuxcnc-uspace
```

3.7.2 Installing on Ubuntu Precise

1. Install Ubuntu Precise 12.04 x86 (32-bit). Any flavor should work (regular Ubuntu, Xubuntu, Lubuntu, etc). 64-bit (AMD64) is currently not supported. You can download the installer here: http://releases.ubuntu.com/precise/
2. Run the following to bring the machine up to date with the latest packages in Ubuntu Precise.

```
sudo apt-get update
sudo apt-get dist-upgrade
```

3. Add the LinuxCNC Archive Signing Key to your apt keyring by running

```
sudo apt-key adv --keyserver hkp://keys.gnupg.net --recv-key 3cb9fd148f374fef
```

4. Add a new apt source

```
sudo add-apt-repository "deb http://linuxcnc.org/ precise base 2.7-rtai"
```

5. Fetch the package list from linuxcnc.org.

```
sudo apt-get update
```

6. Install the RTAI kernel and modules by running

```
sudo apt-get install linux-image-3.4-9-rtai-686-pae rtai-modules-3.4-9-rtai-686-pae
```

7. If you want to be able to build LinuxCNC from source using the git repo, also run

```
sudo apt-get install linux-headers-3.4-9-rtai-686-pae
```

8. Reboot, and make sure you boot into the rtai kernel. When you log in, verify that the kernel name is 3.4-9-rtai-686-pae.

```
uname -r
```

9. Run

```
sudo apt-get install linuxcnc
```

Chapter 4

Updating LinuxCNC

This section describes how to upgrade LinuxCNC from version 2.7 to the new version. It assumes that you have an existing 2.7 install that you want to update.

To upgrade LinuxCNC from a version older than 2.7, you have to first upgrade your old install to 2.7, then follow these instructions to upgrade to the new version.

If you do not have an old version of LinuxCNC to upgrade, then you're best off making a fresh install of the new version as described in the section Getting LinuxCNC.

4.1 Upgrade to the new version

The basic idea is to disable the old linuxcnc.org apt sources and add a new linuxcnc.org apt source, then upgrade the LinuxCNC packages.

The details will depend on which platform you're running on. Run `lsb_release -ic` to find this information out:

```
> lsb_release -ic
Distributor ID: Debian
Codename:       wheezy
```

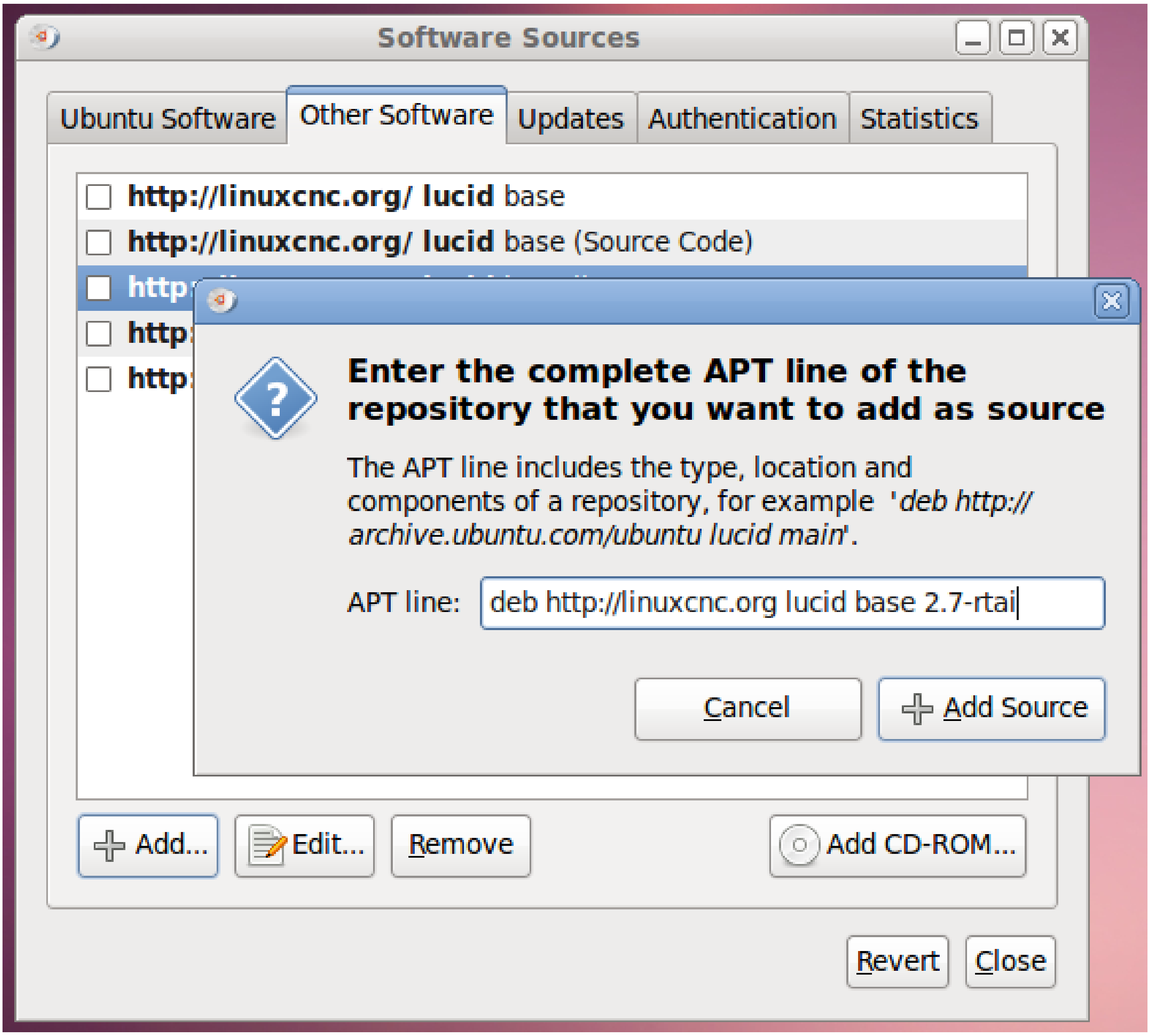
You should be running on Debian Wheezy (as above), or Ubuntu Precise, or Ubuntu Lucid.

4.1.1 Setting apt sources

- Open the `Software Sources` window. The process for doing this differs slightly on the three supported platforms:
  - Debian Wheezy:
    * Click on `Applications Menu`, then `System`, then `Synaptic Package Manager`.
    * In Synaptic, click on the `Settings` menu, then click `Repositories` to open the `Software Sources` window.
  - Ubuntu Precise:
    * Click on the `Dash Home` icon in the top left.
    * In the `Search` field, type "software", then click on the `Ubuntu Software Center` icon.
    * In the Ubuntu Software Center window, click on the `Edit` menu, then click on `Software Sources...` to open the `Software Sources` window.
  - Ubuntu Lucid:
    * Click the `System` menu, then `Administration`, then `Synaptic Package Manager`.
    * In Synaptic, click on the `Settings` menu, then click `Repositories` to open the `Software Sources` window.

- In the `Software Sources` window, select the `Other Software` tab.
- Delete or un-check all the old linuxcnc.org entries (leave all non-linuxcnc.org lines as they are).
- Click the `Add` button and add a new apt line. The line will be slightly different on the different platforms:

Platform	apt source line
Debian Wheezy	`deb http://linuxcnc.org wheezy base 2.7-rtai`
Ubuntu Precise	`deb http://linuxcnc.org precise base 2.7-rtai`
Ubuntu Lucid	`deb http://linuxcnc.org lucid base 2.7-rtai`



- Click `Add Source`, then `Close` in the Software Sources window. If it pops up a window informing you that the information about available software is out-of-date, click the `Reload` button.

4.1.2 Upgrading to the new version

Now your computer knows where to get the new version of the software, next we need to install it.

The process again differs depending on your platform.

4.1.2.1 Debian Wheezy and Ubuntu Lucid

Debian Wheezy and Ubuntu Lucid both use the Synaptic Package Manager.

- Open Synaptic using the instructions in Setting apt sources above.
- Click the `Reload` button.
- Use the Search function to search for `linuxcnc`.
- Click the check box to mark the new linuxcnc and linuxcnc-doc-* packages for upgrade. The package manager may select a number of additional packages to be installed, to satisfy dependencies that the new linuxcnc package has.
- Click the `Apply` button, and let your computer install the new package. The old linuxcnc package will be automatically upgraded to the new one.

4.1.3 Ubuntu Precise

- Click on the `Dash Home` icon in the top left.
- In the `Search` field, type "update", then click on the `Update Manager` icon.
- Click the `Check` button to fetch the list of packages available.
- Click the `Install Updates` button to install the new versions of all packages.

4.2 Updating Configuration Files

The new version of LinuxCNC differs from version 2.7 in some ways that may require changes to your machine configuration.

4.2.1 Distribution Configurations (updates for joints_axes)

The LinuxCNC distribution includes many example configurations organized in directory hierarchies named: by_machine, by_interface, and sim (simulated machines). These configurations are often used as starting points for making a new configuration, as examples for study, or as complete simulated machines that can run without special hardware or real-time kernels.

The configuration files in these directory trees have been updated for the changes required for the joints_axes updates.

4.2.2 Automatic updates (update_ini script for joints_axes)

Since the joints_axes updates require a number of changes to user ini files and their related halfiles, a script named update_ini is provided to automatically convert user configurations.

This script is invoked when a user starts an existing configuration for the first time after updating LinuxCNC. The script searches the user ini file for a [EMC]VERSION item. If this item 1) does not exist, or 2) exists and is set to the historical CVS value "$Revision$", or is a numerical value less than 1.0, then the update_ini script will popup a dialog to offer to edit the user files to create an updated configuration. If the user accepts, the configuration will be updated.

For example, if the user configuration is named bigmill.ini, the bigmill.ini file and its local associated hal files will be edited to incorporate joints_axes changes. All files of the initial configuration will be saved in a new directory named after the original configuration with a ".old" suffix (bigmill.old in the example).

The update_ini script handles all common user items that are found in basic machines employing identity kinematics. Less common items used in more complex machines may not be converted automatically. Examples of complex machine configurations include:

- gantries with two joints for an axis
- machines with jogwheels
- robots with non-identity kinematics
- configurations using haltcl files

The following subsections and the section for *Hal Changes* list items that may require additional user edits to ini or hal files.

4.2.3 Kinematics modules

The gentrivkins and gantrykins kinematics modules have been removed as their functionality is now available in the updated trivkins module.

The gentrivkins module has only been available in prior joints_axes branches. To convert, it is necessary to change the name.

Hal file examples:

```
was: loadrt gentrivkins
 is: loadrt trivkins

was: loadrt gentrivkins coordinates=xyyz
 is: loadrt trivkins    coordinates=xyyz
```

Configurations using gantrykins should be updated to use trivkins with the kinstype= parameter set to BOTH (for KINEMATICS_BOTH).

Hal file example:

```
was: loadrt gantrykins coordinates=xyyz
 is: loadrt trivkins   coordinates=xyyz kinstype=BOTH
```

See the trivkins man page for additional information (*$ man trivkins*)

Note: the most supported usage for specifying kinematics in joints_axes is to set values in the configuration ini file [KINS} section and then reference them within the specified [HAL]HALFILES (.hal .tcl files). For example:

```
inifile:    [KINS]
            KINEMATICS = trivkins
            JOINTS = 3
            ...

halfile:    loadrt [KINS]KINEMATICS

haltclfile: loadrt $::KINS(KINEMATICS)
```

4.2.4 Lathe Configurations

Prior to joints_axes incorporation, lathes were often configured as if they were three axis (XYZ) machines with an unused axis (Y). This was convenient for sharing Hal files (especially for simulation configs) but required specification of [TRAJ]AXES =3, a *dummy* AXIS_Y section, and provisions for homing the unused Y coordinate. These arrangements are no longer required or recommended.

Historical lathe configurations used the default options for the trivkins kinematics module. These default options configure all axis letters (XYZABCUVW). With joints_axes incorporation, a more appropriate kinematics specification sets the coordinates to the exact ones used (XZ) and sets the number of joints accordingly to 2. There is no need for an ini file [AXIS_Y] section and only two [JOINT_N] sections need be defined.

Example ini file items for a lathe (only sections relevant to kinematics are shown):

```
[KINS]
KINEMATICS = trivkins coordinates=xz
JOINTS = 2

[TRAJ]
COORDINATES = XZ
...

[AXIS_X]
...

[AXIS_Z]
...

[JOINT_0]
...

[JOINT_1]
...
```

Note that some simulation configurations may still use the historical lathe configuration precedents.

4.2.5 Consistent Joints/Axes specifications

Ini file items that affect joints and axes usage must be consistent.

The motion kinematics module typically loaded with *[KINS]KINEMATICS=* must use a number of joints equal to the number specified with *[KINS]JOINTS=*.

The kinematics module must implement axis letters that are consistent with the specification used by the task module item *[TRAJ]COORDINATES=*.

Examples:

Three axis Cartesian machine using trivkins (KINEMATICS_IDENTITY):

```
  [KINS]KINEMATICS  = trivkins
  [KINS]JOINTS      = 3
  [TRAJ]COORDINATES = XYZ
```

Two axis lathe using trivkins (KINEMATICS_IDENTITY) with non-consecutive axis letters:

```
  [KINS]KINEMATICS  = trivkins coordinates=XZ
  [KINS]JOINTS      = 2
  [TRAJ]COORDINATES = XZ
```

Gantry using trivkins (KINEMATICS_IDENTITY) with duplicated axis letters:

```
  [KINS]KINEMATICS  = trivkins coordinates=XYYZ
  [KINS]JOINTS      = 4
  [TRAJ]COORDINATES = XYYZ
```

Gantry using trivkins (KINEMATICS_IDENTITY) with duplicated axis letters and a rotary axis with skipped axis letters (A,B skipped):

```
  [KINS]KINEMATICS  = trivkins coordinates=XYYZC
  [KINS]JOINTS      = 5
  [TRAJ]COORDINATES = XYYZC
```

Linear Delta Robot with non-identity kins (KINEMATICS_BOTH) working in Cartesian frame with an additional rotary coordinate:

```
[KINS]KINEMATICS  = lineardeltakins
[KINS]JOINTS      = 4
[TRAJ]COORDINATES = XYZA
```

Note: Some general-purpose kinematics modules (like trivkins) implement identity kinematics with support for coordinate specification (axis letters). Axis letters may be omitted. Axis letters may be duplicated. Joints are assigned to axis letters in a defined manner (*$ man trivkins*).

Note: For trivkins module loading, do not include spaces about the = sign or letters:

```
This:     [KINS]KINEMATICS = trivkins coordinates=XZ
NOT This: [KINS]KINEMATICS = trivkins coordinates = X Z
```

Note: Custom kinematics modules that implement non-identity kinematics (like lineardeltakins) define machine-specific relationships between a set of coordinates and a set of joints. Typically, custom kinematics modules compute the joints-axes relationships within the custom module but it is important to use consistent settings for the related ini items: *[KINS]JOINTS* and *[TRAJ]COORDINATES*. The details will usually be explained in the module man page (for example, *$ man lineardeltakins*).

4.2.6 Home sequences

Negative values may be used for the ini file items named [JOINT_n]HOME_SEQUENCE. Prior to joints_axes incorporation a value of -1 or the ommission of the item indicated no sequence was applicable. Now, only omission of the item is used for that purpose.

4.2.7 Locking rotary indexer (updates for joints_axes)

With joints_axes, an indexer is a joint that can be homed (joint mode) but must also be unlocked from gcode. This requires a one-to-one correspondence between a single joint and an axis.

Specify the joint number that corresponds to a rotary axis (L = A,B, or C) with an ini file setting for the axis:

```
[AXIS_L]LOCKING_INDEXER_JOINT = joint_number_for_indexer
```

Specify that the joint is a locking indexer with an ini file setting for the joint (N is the joint_number_for_indexer):

```
[JOINT_N]LOCKING_INDEXER = 1
```

Hal pins can be created to coordinate use of a locking indicator joint:

```
 joint.N.unlock      (BIT output from Hal)
 joint.N.is-unlocked (BIT input  to   Hal)
```

To create these hal pins for locking joints, specify all joints that are used as locking indexers with the *unlock_joints_mask* parameter for the motmod module. (bit0(LSB)==>joint0, bit1==>joint1, etc.)

```
[EMCMOT]motmod unlock_joints_mask = BITMASK
```

As an example, consider a machine using trivkins kinematics with coordinates XYZB where B is a locking indexer. For trivkins, joint numbers (starting with 0) are assigned consecutively to the coordinates specified (axis coordinate letters may be omitted). For this example, X==>joint0, Y==>joint1, Z==>joint2, B==>joint3. The mask to specify joint 3 is 000001000 (binary) == 0x08 (hexadecimal)

The required ini file entries for this trivkins XYZB example are:

```
[KINS]
JOINTS = 4
KINEMATICS = trivkins coordinates=XYZB
...

[TRAJ]
COORDINATES = XYZB
...

[EMCMOT]
EMCMOT = motmod unlock_joints_mask = 0x08
...

[AXIS_B]
LOCKING_INDEXER_JOINT = 3
...

[JOINT_3]
LOCKING_INDEXER = 1
...
```

For more complex kinematics, select the joint number as required — there must be a one-to-one correspondence between the rotary axis and the joint number.

(See the motion man page (*$ man motion*) for more information on motmod)

4.2.8 Stricter INI file syntax

Lines with numeric INI variables are no longer allowed to have trailing text. In earlier versions of LinuxCNC any text after the number was silently ignored, but as of this version such text is totally disallowed. This includes hash characters ("#"), which in this position are a part of the value, not a comment character.

For example, lines like this will no longer be accepted:

```
MAX_VELOCITY = 7.5 # This is the max velocity of the axis.
```

They could be transformed into pairs of lines like this:

```
# This is the max velocity of the axis.
MAX_VELOCITY = 7.5
```

4.3 Hal Changes (updates for joints_axes)

4.3.1 Wheel or MPG (manual pulse generator) jogging

Prior to incorporation of joints_axes updates, wheel jogging was supported in joint mode only and controlled with hal pins:

```
bit   IN  axis.M.jog-enable
float IN  axis.M.jog-scale
s32   IN  axis.M.jog-counts
bit   IN  axis.M.jog-vel-mode
```

where *M* is a number corresponding to an axis letter (0==>X, 1==>Y, etc.)

With incorporation of joints_axes updates, wheel jogging is available for joints in joint mode and for each axis coordinate in teleop mode. The controlling hal pins provided are:

```
bit   IN  joint.N.jog-enable
float IN  joint.N.jog-scale
s32   IN  joint.N.jog-counts
bit   IN  joint.N.jog-vel-mode

bit   IN  axis.L.jog-enable
float IN  axis.L.jog-scale
s32   IN  axis.L.jog-counts
bit   IN  axis.L.jog-vel-mode
```

where *N* is a joint number and *L* is an axis letter.

To use an MPG in identity kins configurations where there is a one-to-one correspondence of a joint number and an axis letter, it may be convenient to connect the corresponding hal pins. For example, if joint 1 corresponds exactly to axis letter y:

```
net jora_1_y_enable   => joint.1.jog-enable => axis.y.jog-enable
net jora_1_y_scale    => joint.1.jog-scale  => axis.y.jog-scale
net jora_1_y_counts   => joint.1.jog-counts => axis.y.jog-counts
net jora_1_y_vel-mode => joint.1.jog-counts => axis.y.jog-vel-mode
```

(The signal names jora_1_y_* are examples, names prior to conversion for joints_axes will depend upon the specific configuration details.)

Configurations with non-identity kinematics and configurations that use duplicated axis letters (for example, gantries using more than one joint for an axis coordinate) will require appropriate independent control logic to support both joint and teleop (world) jogging.

4.3.2 Ini Hal pins

Hal pins are created for ini file items for both joints ([JOINT_N] stanzas) and axes ([AXIS_L] stanzas):

```
For N = 0 ... [KINS](JOINTS -1)
Ini File Item              hal pin name
[JOINT_N]BACKLASH          ini.N.backlash
[JOINT_N]FERROR            ini.N.ferror
[JOINT_N]MIN_FERROR        ini.N.min_ferror
[JOINT_N]MIN_LIMIT         ini.N.min_limit
[JOINT_N]MAX_LIMIT         ini.N.max_limit
[JOINT_N]MAX_VELOCITY      ini.N.max_velocity
[JOINT_N]MAX_ACCELERATION  ini.N.max_acceleration
[JOINT_N]HOME              ini.N.home
[JOINT_N]HOME_OFFSET       ini.N.home_offset

For L = x y z a b c u v w:
Ini File Item              hal pin name
[AXIS_L]MIN_LIMIT          ini.L.min_limit
[AXIS_L]MAX_LIMIT          ini.L.max_limit
[AXIS_L]MAX_VELOCITY       ini.L.max_velocity
[AXIS_L]MAX_ACCELERATION   ini.L.max_acceleration
```

Note: In prior versions of LinuxCNC (before joints_axes updates), the hal pin names *ini.N.** referred to axes with 0==>x, 1==>y, etc. (pins were created for all 9 axes) See the man page (*$ man milltask*) for more information

4.4 GUIs (updates for joints_axes)

4.4.1 Halui

Halui now supports teleop jogging resulting in some changed pin names and numerous new names for jogging-related pins.

See the man page (*$ man halui*) for all pin names.

4.4.1.1 TELEOP jogging (also called axis or world jogging)

New pins for teleop jogging are:

```
      new: halui.axis.jog-speed
      new: halui.axis.jog-deadband

      new: halui.axis.L.plus
      new: halui.axis.L.minus
           ... etc.
```

where *L* is a letter corresponding to one of the axis letters specified by [TRAJ]COORDINATES or *selected* for the axis selected by the halui.axis.L.select pins.

4.4.1.2 Joint jogging

All pins for joint jogging were renamed for specificity:

```
      was: halui.jog-speed          is: halui.joint.jog-speed
      was: halui.jog-deadband       is: halui.joint.jog-deadband

      was: halui.jog.N.plus         is: halui.joint.N.plus
      was: halui.jog.N.minus        is: halui.joint.N.minus
           ...  etc.                    ... etc.
```

where *N* is a joint number (0 … num_noints-1) or *selected* for the joint selected by the halui.joint.N.select pins.

4.4.1.3 Aditional pin renames

The hal pins for *selected* joints were renamed for consistency with related pins.

```
      was: halui.joint.selected.is_homed
       is: halui.joint.selected.is-homed

      was: halui.joint.selected.on-soft-limit
       is: halui.joint.selected.on-soft-min-limit
```

4.4.2 Axis gui

4.4.2.1 Kinematics Support

The axis gui continues to support identity kinematics configurations. This gui hides the distinctions of axes and joints in order to simplify the display and usage of simple machines.

Some machines, typically gantrys, may use a configuration with more than one joint assigned to an axis letter. This can be done with the trivkins kinematics module using repeated coordinate letters. For example, a machine configured with ini settings:

```
[KINS]
KINEMATICS = trivkins coordinates=XYYZ kinstype=BOTH
...
[TRAJ]
COORDINATES = XYYZ
...
```

This machine, after homing, has a one-to-one correspondence between a single axis letter (Y) and a pair of joints (1,2). Using *kinematics=BOTH* allows control of individual joints in joint mode *if/when required*.

The axis gui supports configurations using non-identity kinematics with

1. Key binding (*$*) to toggle joint or teleop mode
2. Preview Tab display of joints or axes according to joint or teleop mode
3. Preview Tab display of *Home* and *Limit* icons in joint mode
4. Preview Tab display of *All-homed* and 'Any-limit icons in teleop mode
5. DRO Tab display of joint or axes according to joint or teleop mode
6. Provision to automatically switch to teleop mode after homing (see below for AUTO_TELEOP)

4.4.2.2 AUTO_TELEOP

The axis gui supports an option, [KINS]AUTO_TELEOP = N_seconds, to cause an automatic switch to teleop mode after homing. The N_seconds parameter specifes the maximum time to wait after initiating homing with the Home-All button. The setting is applicable only for kinematics that are not KINEMATICS_IDENTITY.

Usage Example (wait 20 seconds max):

```
[KINS]
AUTO_TELEOP = 20
...
```

4.4.2.3 Home icons

For identity kinematics, *Home* icons are shown for the correspoinding (one-to-one) axis letter when a joint is homed.

For non-identity kinematics, *Home* icons are shown for individual joints when a joint is homed in joint display mode. An *All-homed* icon is displayed for all axis letters when ALL joints are homed in world display mode.

4.4.2.4 Limit icons

For identity kinematics, *Limit* icons are shown for the corresponding (one-to-one) axis letter when a joint limit is active.

For non-identity kinematics, *Limit* icons are shown for individual joints when the joint limit is active in joint display mode. An *Any-Limit* icon is displayed if any joint is at a limit in teleop display mode.

4.4.2.5 Key bindings for a fourth axis

In the AXIS gui, jogging keys are assigned to axes in a configurable fashion. For 3-axis machines, XYZA machines, and lathes the default is the same as in 2.7. For other machines, the 4 pairs of jogging keys are assigned to the first 4 axes that exist in the order XYZ ABC UVW. These assignments can be controlled by new inifile directives in the [DISPLAY] section of the inifile

4.4.3 tklinuxcnc

4.4.3.1 Kinematics Support

The tklinuxcnc gui supports both identity and non-identity kinematics, includes gui radiobuttons and a key bindiing (*$*) for toggling joint and teleop modes. Jogging is supported in both joint and teleop modes. Note that the values used for jogging may not be appropriate for both modes.

4.4.4 touchy

The touchy gui continues to support the identity kinematics configurations that it supported prior to joints_axes incorporation. Jogging is done in joint mode (no coordination of joint/teleop modes with other guis is currently provided).

4.4.5 gscreen

The gscreen gui continues to support the identity kinematics configurations that it supported prior to joints_axes incorporation. Jogging is done in joint mode by default with limited support for teleop jogging when set by a contemporaneous, cooperating gui. Note that the values used for jogging may not be appropriate for both modes.

4.4.6 gmoccapy

The gmoccapy gui continues to support the identity kinematics configurations that it supported prior to joints_axes incorporation. Jogging is done in joint mode (no coordination of joint/teleop modes with other guis is currently provided).

4.4.7 linuxcnclcd

The linuxcnclcd gui has been updated to continue support for the identity kinematics supported prior to joints_axes incorporations. Jogging is done in joint mode. Compatiblity testsing awaits users with the required hardware.

4.4.8 linuxcncrsh

The jogging commands have been altered to accomodate both joint (free) and teleop (world) jogging.

```
    was: set jog      joint_number              speed
     is: set jog      joint_number|axis_letter speed

    was: set jog_incr joint_number              speed increment
     is: set jog_incr joint_number|axis_letter speed increment

    was: set jog_stop
     is: set jog_stop joint_number|axis_letter
```

Note: Test for teleop mode using command: get teleop_enable if TELEOP_ENABLE=YES, use axis_letter else use joint_number

Note: Formerly, the command *set jog 0 1.234* would jog the zeroth axis (X) with requested speed=1.234 in any mode (free or teleop). This command now attemps to jog the zeroth joint (Joint0) provided the mode is free (not teleop). To jog the X axis, the mode must be teleop and the corresponding command is: *set jog x 1.234*

4.5 Simulator configurations (updates for joints axes)

4.5.1 Pre-joints_axes

Prior to joints_axes incorporation, the halfiles used in sim configs typically supported a common milling machine — a Cartesian system with trivial kinematics and three axes named *X Y Z*. Typical halfile entries:

```
[HAL]
HALFILE = core_sim.hal
HALFILE = sim_spindle_encoder.hal
HALFILE = axis_manualtoolchange.hal
HALFILE = simulated_home.hal
```

Lathe configs often shared the same halfiles and used the expedient method of specifying 3 axes with *Y* unused. More complex sim configs provided specific sets of halfiles according to the configuration purpose.

4.5.2 Post-joints_axes

With the incorporation of joints_axes functionality, may sims provided in the distribution now take advantage of a general purpose halfile that supports numerous configurations automatically. A typical sim config HALFILE specification is:

```
[HAL]
HALFILE = LIB:basic_sim.tcl
```

The basic_sim.tcl HALFILE supports a number of commonly required functions for any number of joints as specified by:

```
[KINS]
...
JOINTS = number_of_joints
...
```

Functions supported include:

1. *ddts* — differentiator hal components are loaded and connected for each joint (and xy, xyz for trivkins machines)
2. *simulated_home* — a sim_home_switch hal component is loaded and connected for each joint. The homing conditions are specified by the usual [JOINT_n]HOME_* ini file items.
3. *use_hal_manualtoolchange* — the user space hal_manualtoolchange component is loaded and connected.
4. *sim_spindle* — the sim_spindle component is loaded and connected to additional loaded hal components to simulate the inertia of a rotating spindle mass.

These functions are activated by default but can be excluded using options *-no_make_ddts, -no_simulated_home, -no_use_hal_manualtoo* *-no_sim_spindle*. For example, to omit ddts:

```
HALFILE = LIB:basic_sim.tcl -no_make_ddts
```

Omitting one or more of the core functions allow a user to test without the function or add new HALFILEs to implement or expand on the functionality.

4.5.3 Notes

All components and connections made by LIB:basic_sim.tcl can be viewed using halcmd. The entire hal configuration (except for userspace components loaded with loadusr) can be saved to a file using:

```
$ halcmd save > hal.save
```

Use of LIB:basic_sim.tcl reduces the effort needed to make a simulation config since it handles most of the required component loading and hal connections.

The sim config *Sample Configurations/sim/axis/minimal_xyz.ini* demonstrates a working xyz configuration that uses LIB:basic_sim.tcl with a minimal number of ini file settings.

Chapter 5

Glossary

A listing of terms and what they mean. Some terms have a general meaning and several additional meanings for users, installers, and developers.

Acme Screw
A type of lead-screw that uses an Acme thread form. Acme threads have somewhat lower friction and wear than simple triangular threads, but ball-screws are lower yet. Most manual machine tools use acme lead-screws.

Axis
One of the computer controlled movable parts of the machine. For a typical vertical mill, the table is the X axis, the saddle is the Y axis, and the quill or knee is the Z axis. Angular axes like rotary tables are referred to as A, B, and C. Additional linear axes relative to the tool are called U, V, and W respectively.

Axis(GUI)
One of the Graphical User Interfaces available to users of LinuxCNC. It features the modern use of menus and mouse buttons while automating and hiding some of the more traditional LinuxCNC controls. It is the only open-source interface that displays the entire tool path as soon as a file is opened.

Gmoccapy (GUI)
A Graphical User Interfaces available to users of LinuxCNC. It features the use and feel of an industrial comtrol and can be used with touch screen, mouse and keyboard. It support embedded tabs and hal driven user messages, it offers a lot of hal beens to be controled with hardware. Gmoccapy is highly cusomizable.

Backlash
The amount of "play" or lost motion that occurs when direction is reversed in a lead screw. or other mechanical motion driving system. It can result from nuts that are loose on leadscrews, slippage in belts, cable slack, "wind-up" in rotary couplings, and other places where the mechanical system is not "tight". Backlash will result in inaccurate motion, or in the case of motion caused by external forces (think cutting tool pulling on the work piece) the result can be broken cutting tools. This can happen because of the sudden increase in chip load on the cutter as the work piece is pulled across the backlash distance by the cutting tool.

Backlash Compensation
Any technique that attempts to reduce the effect of backlash without actually removing it from the mechanical system. This is typically done in software in the controller. This can correct the final resting place of the part in motion but fails to solve problems related to direction changes while in motion (think circular interpolation) and motion that is caused when external forces (think cutting tool pulling on the work piece) are the source of the motion.

Ball Screw
A type of lead-screw that uses small hardened steel balls between the nut and screw to reduce friction. Ball-screws have very low friction and backlash, but are usually quite expensive.

Ball Nut
A special nut designed for use with a ball-screw. It contains an internal passage to re-circulate the balls from one end of the screw to the other.

CNC
Computer Numerical Control. The general term used to refer to computer control of machinery. Instead of a human operator turning cranks to move a cutting tool, CNC uses a computer and motors to move the tool, based on a part program.

Comp
A tool used to build, compile and install LinuxCNC HAL components.

Configuration(n)
A directory containing a set of configuration files. Custom configurations are normally saved in the users home/linuxcnc/-configs directory. These files include LinuxCNC's traditional INI file and HAL files. A configuration may also contain several general files that describe tools, parameters, and NML connections.

Configuration(v)
The task of setting up LinuxCNC so that it matches the hardware on a machine tool.

Coordinate Measuring Machine
A Coordinate Measuring Machine is used to make many accurate measurements on parts. These machines can be used to create CAD data for parts where no drawings can be found, when a hand-made prototype needs to be digitized for moldmaking, or to check the accuracy of machined or molded parts.

Display units
The linear and angular units used for onscreen display.

DRO
A Digital Read Out is a system of position-measuring devices attached to the slides of a machine tool, which are connected to a numeric display showing the current location of the tool with respect to some reference position. DROs are very popular on hand-operated machine tools because they measure the true tool position without backlash, even if the machine has very loose Acme screws. Some DROs use linear quadrature encoders to pick up position information from the machine, and some use methods similar to a resolver which keeps rolling over.

EDM
EDM is a method of removing metal in hard or difficult to machine or tough metals, or where rotating tools would not be able to produce the desired shape in a cost-effective manner. An excellent example is rectangular punch dies, where sharp internal corners are desired. Milling operations can not give sharp internal corners with finite diameter tools. A *wire* EDM machine can make internal corners with a radius only slightly larger than the wire's radius. A *sinker* EDM can make internal corners with a radius only slightly larger than the radius on the corner of the sinking electrode.

EMC
The Enhanced Machine Controller. Initially a NIST project. Renamed to LinuxCNC in 2012.

EMCIO
The module within LinuxCNC that handles general purpose I/O, unrelated to the actual motion of the axes.

EMCMOT
The module within LinuxCNC that handles the actual motion of the cutting tool. It runs as a real-time program and directly controls the motors.

Encoder
A device to measure position. Usually a mechanical-optical device, which outputs a quadrature signal. The signal can be counted by special hardware, or directly by the parport with LinuxCNC.

Feed
Relatively slow, controlled motion of the tool used when making a cut.

Feed rate
The speed at which a cutting motion occurs. In auto or mdi mode, feed rate is commanded using an F word. F10 would mean ten machine units per minute.

Feedback
A method (e.g., quadrature encoder signals) by which LinuxCNC receives information about the position of motors

Feedrate Override
A manual, operator controlled change in the rate at which the tool moves while cutting. Often used to allow the operator to adjust for tools that are a little dull, or anything else that requires the feed rate to be “tweaked”.

Floating Point Number
A number that has a decimal point. (12.300) In HAL it is known as float.

G-Code
The generic term used to refer to the most common part programming language. There are several dialects of G-code, LinuxCNC uses RS274/NGC.

GUI
Graphical User Interface.

General
A type of interface that allows communications between a computer and a human (in most cases) via the manipulation of icons and other elements (widgets) on a computer screen.

LinuxCNC
An application that presents a graphical screen to the machine operator allowing manipulation of the machine and the corresponding controlling program.

HAL
Hardware Abstraction Layer. At the highest level, it is simply a way to allow a number of building blocks to be loaded and interconnected to assemble a complex system. Many of the building blocks are drivers for hardware devices. However, HAL can do more than just configure hardware drivers.

Home
A specific location in the machine’s work envelope that is used to make sure the computer and the actual machine both agree on the tool position.

ini file
A text file that contains most of the information that configures LinuxCNC for a particular machine.

Instance
One can have an instance of a class or a particular object. The instance is the actual object created at runtime. In programmer jargon, the Lassie object is an instance of the Dog class.

Joint Coordinates
These specify the angles between the individual joints of the machine. See also Kinematics

Jog
Manually moving an axis of a machine. Jogging either moves the axis a fixed amount for each key-press, or moves the axis at a constant speed as long as you hold down the key. In manual mode, jog speed can be set from the graphical interface.

kernel-space
See real-time.

Kinematics
The position relationship between world coordinates and joint coordinates of a machine. There are two types of kinematics. Forward kinematics is used to calculate world coordinates from joint coordinates. Inverse kinematics is used for exactly the opposite purpose. Note that kinematics does not take into account, the forces, moments etc. on the machine. It is for positioning only.

Lead-screw
An screw that is rotated by a motor to move a table or other part of a machine. Lead-screws are usually either ball-screws or acme screws, although conventional triangular threaded screws may be used where accuracy and long life are not as important as low cost.

Machine units
The linear and angular units used for machine configuration. These units are specified and used in the ini file. HAL pins and parameters are also generally in machine units.

MDI
Manual Data Input. This is a mode of operation where the controller executes single lines of G-code as they are typed by the operator.

NIST
National Institute of Standards and Technology. An agency of the Department of Commerce in the United States.

NML
Neutral Message Language provides a mechanism for handling multiple types of messages in the same buffer as well as simplifying the interface for encoding and decoding buffers in neutral format and the configuration mechanism.

Offsets
An arbitrary amount, added to the value of something to make it equal to some desired value. For example, gcode programs are often written around some convenient point, such as X0, Y0. Fixture offsets can be used to shift the actual execution point of that gcode program to properly fit the true location of the vise and jaws. Tool offsets can be used to shift the "uncorrected" length of a tool to equal that tool's actual length.

Part Program
A description of a part, in a language that the controller can understand. For LinuxCNC, that language is RS-274/NGC, commonly known as G-code.

Program Units
The linear and angular units used in a part program. The linear program units do not have to be the same as the linear machine units. See G20 and G21 for more information. The angular program units are always measured in degrees.

Python
General-purpose, very high-level programming language. Used in LinuxCNC for the Axis GUI, the Stepconf configuration tool, and several G-code programming scripts.

Rapid
Fast, possibly less precise motion of the tool, commonly used to move between cuts. If the tool meets the workpiece or the fixturing during a rapid, it is probably a bad thing!

Rapid rate
The speed at which a rapid motion occurs. In auto or mdi mode, rapid rate is usually the maximum speed of the machine. It is often desirable to limit the rapid rate when testing a g-code program for the first time.

Real-time
Software that is intended to meet very strict timing deadlines. Under Linux, in order to meet these requirements it is necessary to install a realtime kernel such as RTAI and build the software to run in the special real-time environment. For this reason real-time software runs in kernel-space.

RTAI
Real Time Application Interface, see https://www.rtai.org/, the real-time extensions for Linux that LinuxCNC can use to achieve real-time performance.

RTLINUX
See https://en.wikipedia.org/wiki/RTLinux, an older real-time extension for Linux that LinuxCNC used to use to achieve real-time performance. Obsolete, replaced by RTAI.

RTAPI
A portable interface to real-time operating systems including RTAI and POSIX pthreads with realtime extensions.

RS-274/NGC
The formal name for the language used by LinuxCNC part programs.

Servo Motor
Generally, any motor that is used with error-sensing feedback to correct the position of an actuator. Also, a motor which is specially-designed to provide improved performance in such applications.

Servo Loop
A control loop used to control position or velocity of an motor equipped with a feedback device.

Signed Integer
A whole number that can have a positive or negative sign. In HAL it is known as s32. (A signed 32-bit integer has a usable range of -2,147,483,647 to +2,147,483,647.)

Spindle
The part of a machine tool that spins to do the cutting. On a mill or drill, the spindle holds the cutting tool. On a lathe, the spindle holds the workpiece.

Spindle Speed Override
A manual, operator controlled change in the rate at which the tool rotates while cutting. Often used to allow the operator to adjust for chatter caused by the cutter's teeth. Spindle Speed Override assumes that the LinuxCNC software has been configured to control spindle speed.

Stepconf
An LinuxCNC configuration wizard. It is able to handle many step-and-direction motion command based machines. It writes a full configuration after the user answers a few questions about the computer and machine that LinuxCNC is to run on.

Stepper Motor
A type of motor that turns in fixed steps. By counting steps, it is possible to determine how far the motor has turned. If the load exceeds the torque capability of the motor, it will skip one or more steps, causing position errors.

TASK
The module within LinuxCNC that coordinates the overall execution and interprets the part program.

Tcl/Tk
A scripting language and graphical widget toolkit with which several of LinuxCNCs GUIs and selection wizards were written.

Traverse Move
A move in a straight line from the start point to the end point.

Units
See "Machine Units", "Display Units", or "Program Units".

Unsigned Integer
A whole number that has no sign. In HAL it is known as u32. (An unsigned 32-bit integer has a usable range of zero to 4,294,967,296.)

World Coordinates
This is the absolute frame of reference. It gives coordinates in terms of a fixed reference frame that is attached to some point (generally the base) of the machine tool.

Chapter 6

Legal Section

6.1 Copyright Terms

6.2 GNU Free Documentation License

GNU Free Documentation License Version 1.1, March 2000

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary

Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five). C. State on the Title page the name of the publisher of the Modified Version, as the publisher. D. Preserve all the copyright notices of the Document. E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. H. Include an unaltered copy of this License. I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence. J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission. K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein. L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version. N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Chapter 7

Index

A
acme screw, 21
axis, 21

B
backlash, 21
backlash compensation, 21
ball nut, 21
ball screw, 21

C
CNC, 22
comp, 22
coordinate measuring machine, 22

D
display units, 22
DRO, 22

E
EDM, 22
EMC, 22
EMCIO, 22
EMCMOT, 22
encoder, 22

F
feed, 22
feed rate, 22
feedback, 22
feedrate override, 23

G
G-Code, 23
Getting LinuxCNC, 3
GUI, 21, 23

H
HAL, 23
home, 23

I
INI, 23
Instance, 23

J
jog, 23
joint coordinates, 23

K
kinematics, 23

L
lead screw, 23
loop, 24

M
machine units, 23
MDI, 24
Minimum Requirements, 3

N
NIST, 24
NML, 24

O
offsets, 24

P
part Program, 24
program units, 24

R
rapid, 24
rapid rate, 24
real-time, 24
RS274NGC, 24
RTAI, 24
RTAPI, 24
RTLINUX, 24

S
servo motor, 24
Signed Integer, 25
spindle, 25
stepper motor, 25
System Requirements, 2

T
TASK, 25

Tk, 25
Traverse Move, 25

U
units, 25
Unsigned Integer, 25
Updates to LinuxCNC, 6
Updating LinuxCNC, 8

W
world coordinates, 25