

Procedure for installing NCam and embedding it in LinuxCNC Debian Wheezy. Note: Since support for Debian Wheezy ended May 2018 I discovered that many sudo apt-get install commands described in many Ncam installation instructions no longer work, so this is the procedure I used to install NCam.

1. Installation

1. Clone NCam from <https://github.com/fern/nativecam>.

2. Make sure python-lxml installed. To verify, open a terminal and type python

If python is not installed, clone it from a website. Note: The following terminal command will not work for installation now that Debian Wheezy is no longer supported.

```
sudo apt-get install python-lxml
```

2. Testing Stand Alone

To start, find the file Ncam.py and double click it.

3. To Embed Ncam in LinuxCNC

1. Install python-lxml if not done yet.

2. Using the steps below, create links into /usr/share/pyshared/gladevc/

1. Open a terminal window and type
cd /usr/share/pyshared/gladevc/

2. Type in the open terminal
sudo ln /HOME/fernand(replace w/your user name)/ncam/ncam.py -s
sudo ln /HOME/fernand(replace w/your user name)/ncam/ncam.glade -s

3. Using the steps below, modify hal_pythonplugin.py in /usr/share/pyshared/gladevc/

1. Option 1: If you have gedit,
 1. Open a terminal and type
sudo gedit /usr/share/pyshared/gladevc/hal_pythonplugin.py
 2. Add this new line anywhere and save the file:
import ncam

2. Option 2: If you DO NOT have gedit,
 1. Open a terminal and type
cd /usr/share/pyshared/gladevc/
 2. In the terminal type
sudo nano hal_pythonplugin.py
 3. Add this new line anywhere and save the file:
import ncam

3. Modify hal_python.xml in /usr/share/glade3/catalogs (/glade3 can be glade2)

1. Open a terminal and type

```
sudo gedit /usr/share/glade3/catalogs/hal_python.xml
```

2. Find (it is in the beginning):

```
<glade-widget-classes>
```

Add after:

```
<glade-widget-class name="Ncam" generic-name="ncam" title="ncam">
```

```
<properties>
```

```
<property id="size" query="False" default="1" visible="False"/>
```

```
<property id="spacing" query="False" default="0" visible="False"/>
```

```
<property id="homogeneous" query="False" default="0" visible="False"/>
```

```
</properties>
```

```
</glade-widget-class>
```

3. Find:

```
<glade-widget-group name="python" title="HAL Python">
```

Add after :

```
<glade-widget-class-ref name="Ncam"/>
```

IMPORTANT NOTE : when linuxcnc updates, it recreates directories and if features do not load you will have to redo steps 2, 3 and 4

4. Using the steps below, create links into /usr/lib/pymodules/python2.7

1. Open a terminal window and type

```
cd /usr/lib/pymodules/python2.7/gladevcp
```
2. Type in the open terminal

```
sudo ln /HOME/fernand(replace w/your user name)/ncam/ncam.py -s  
sudo ln /HOME/fernand(replace w/your user name)/ncam/ncam.glade -s
```

4. Using Embedded

-
1. Add these lines into your .ini file inside [DISPLAY] section :

```
# required NativeCAM item:  
NCAM_DIR = ncam  
  
# required NativeCAM item:  
PROGRAM_PREFIX = /home/ fernand(replace w/your user name)/linuxcnc/nc_files  
  
# required NativeCAM item:  
GLADEVCP = -U --catalog=lathe-mm ncam.ui  
NCAM_PATH = /home/fernand(replace w/your user name)/ncam  
  
To use different features use one of the catalogs  
--catalog=lathe  
--catalog=lathe-mm  
--catalog=mill  
--catalog=mill-mm
```

2. Add this line into your .ini file inside [RS274NGC] section :
Required NativeCAM item:
SUBROUTINE_PATH = ncam/my-stuff:ncam/lib/lathe:ncam/lib/utilities

5. Optional Translations

Translation will work in Stand Alone AND Embedded modes

Make links in your system locale directories to translation files

```
cd /usr/share/locale/<YOUR LOCALE>/LC_MESSAGES
```

```
sudo ln /<full path to features directory>/locale/<YOUR LOCALE>/LC_MESSAGES/ncam.mo -s
```

Use poedit to translate strings in ncam.po then save and copy ncam.mo to above path.

6. Extending subroutines

1. Param substitutions

"#param_name" can be used to substitute parameters from the feature.

"#self_id" - unique id made of feature Name + smallest integer id.

2. Eval and exec

```
<eval>"hello world!"</eval>
```

everything inside <eval> tag will be passed

trought python's eval function.

```
<exec>print "hello world"</exec>
```

almost the same but will take all printed data.

you can use self as feature's self.

3. Including Gcode

G-code files can be included by using one of the following functions:

```
[DEFINITIONS]
```

```
content =
```

```
    <eval>self.include_once("rotate-xy.ngc")</eval>
```

```
    <eval>self.include("some-include-file.inc")</eval>
```

4. Data types

[SUBROUTINE] type should be lower case, short, without space. Ex : circle, rect, probe-dn

Valid params types are : string, float, int, bool (or boolean), header (or hdr), combo, items

Note : you can change string, float and int types on the fly with the context menu.

This is usefull with variables.

When using a value like #<var_name> use "string" because if will evaluate to 0 if "int" used or 0.0 if "float".

Study examples in ini/mill/fv_circle.ini and others.

Note : icons and images only need name.ext.

Some information in Russian can be obtained here: <http://cnc-club.ru/forum/viewtopic.php?f=15&t=3124&p=72441#p72441>