



Gcodetools v6.1

(plug-in para inkscape)

manual de usuario

por

Iñigo Zuluaga

[txapuzas.
blogspot.com](http://txapuzas.blogspot.com)

Gcodetools

Gcodetools es un plug-in para Inkscape. Prepara y convierte trayectos desde Inkscape a Gcode, usando interpolation biarco.

Tipo	Plug-in del editor vectorial Inkscape
Desarrollador	Grupo de Desarrollo (Nick Drobchenko, Vladimir Kalyaev, John Brooker, Henry Nicolas)
Escrito en	Python
OS	Varias plataformas (Windows, Linux, MacOS)
Versión	1.6.03
Licencia	GNU GPL
Descargas	5500+
Web	http://cnc-club.ru/forum/viewtopic.php?t=35

Características

Exportar a Gcode

- Exporta trayectos a Gcode
- Utiliza interpolación circular (mediante aproximación biarco) o lineal
- Subdivisión automática del trayecto para llegar a la tolerancia definida
- Procesamiento de múltiples herramientas
- Exportación de Gcode en forma paramétrica y de forma plana
- Personalización de encabezados y pies de página automáticos
- Selección de las unidades
- Procesamiento Multi-paso
- Sufijo numerado automático en los archivos generados para evitar la sobre escritura

Torno

- Cálculo de trayectorias para torno
- Pasadas de acabado
- Definición de la profundidad de pasadas de acabado
- Definición de pasadas de acabado
- Dos funciones de cálculo diferente para pasadas finas
- Reasignación de los ejes estandard

Procesado de Zonas

- Generación de trayectos de zona
- Los trayectos de la zona se pueden modificar

Grabado

- Generación de la trayectoria en función de la forma de la fresa
- Definición de diferentes formas de fresa

Biblioteca de herramientas

- Definición de parámetros de la herramienta (diámetro, feed-avance, el paso de profundidad, avance de penetración, Gcode modificable antes y después de cada trayecto, forma de las fresas,...)
- Las herramientas pueden ser gestionadas mediante procedimientos estándar de Inkscape (copiar, eliminar, asignar a una capa diferente)
- Procesado para múltiples herramientas

Orientación del sistema

- Escala a lo largo de cualquiera de los ejes
- Giro en el plano XY
- Desplazamiento a lo largo de cualquiera de los ejes
- Transformación de acuerdo a puntos arbitrarios

Post-procesador

- Puede crear post-procesadores escribiendo los comandos o seleccionar alguno definido de la lista de post-procesadores por defecto
- Escalado y desplazamiento del Gcode
- Comandos de reasignación Gcode
- Parametrización Gcode
- Redondeo de los valores de coma flotante a la precisión especificada

Verificación de herramientas

- Selección y eliminación de pequeños trayectos (Útiles de zona)
- Verificación de la alineación de la herramienta
- Verificación del sentido de corte

Plotter de corte

- Exportación a Gcode para plotter con cuchilla tangencial. El cuarto eje de rotación es la cuchilla de corte.

Instalación

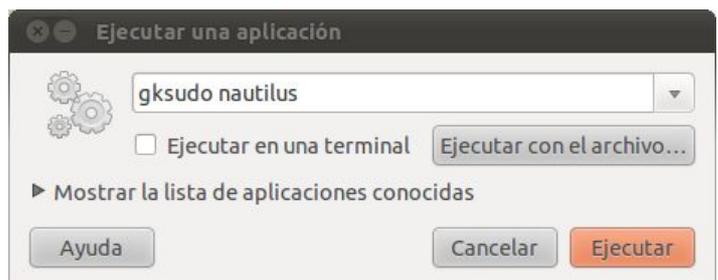
Windows

Es necesario descomprimir y copiar todos los archivos en el directorio:
Archivos de programa\Inkscape\share\extensions\
y después reiniciar inkscape

Linux

Es necesario descomprimir y copiar todos los archivos en el directorio
usr/share/inkscape/extensions/
y después reiniciar inkscape

Para copiar los archivos son necesarios permisos de administrador:
Al pulsar Alt+F2 y aparecerá la pantalla de la derecha, donde hay que teclear **gksudo nautilus**. Esto abrirá Nautilus con permisos de administrador (siempre que introduzcas la contraseña claro!) y ya puedes copiar y pegar los archivos.



Al reiniciar inkscape te aparecerá un nuevo item en Menú/Extensiones/Gcodetools

Puedes acceder a "Todo en uno" con todas las opciones en una ventana, o a cada opción individual. Normalmente tendremos una lengüeta activa (que reacciona al botón situado en la parte inferior izquierda) que será la primera, y otras lengüetas auxiliares, normalmente de definición de parámetros, que variarán en función de las necesidades.

Descripción

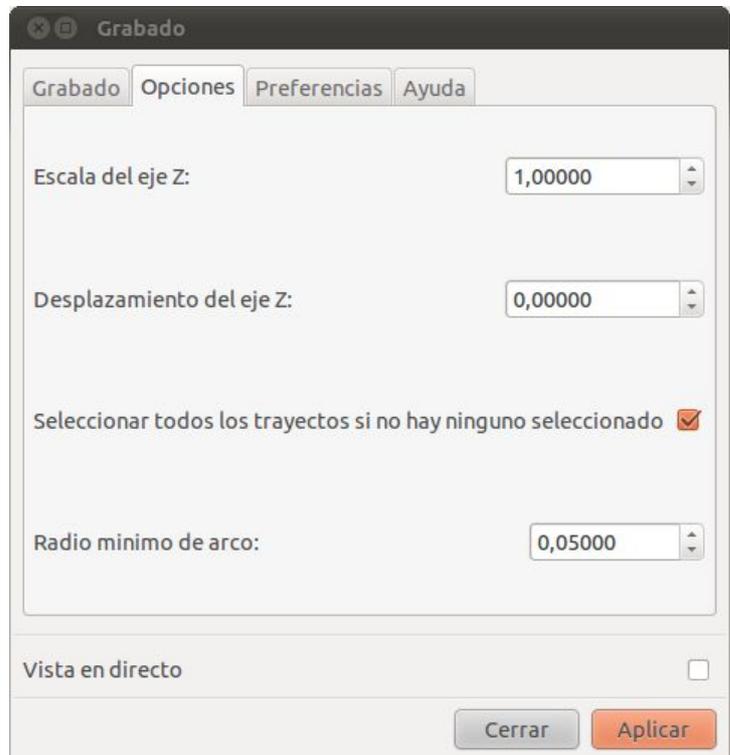
Veamos cada una de las lengüetas, primeramente las comunes:

Opciones

Escala del Eje Z y Desplazamiento del eje Z

Hacen que el gcode del programa tenga modificado respectivamente la escala del eje Z y /o un desplazado el origen del eje Z. Es decir, podemos por ejemplo, generar varios programas Gcode , modificando el Desplazamiento del Eje Z para ir mecanizando diferentes capas.

El **Radio mínimo de arco** es el valor mínimo a partir del cual el programa considerará el arco como una línea. EMC2 genera un error si el radio del arco es menor a 0,2mm, por lo que este valor ha de ser mayor (Si las unidades están en milímetros)



Preferencias

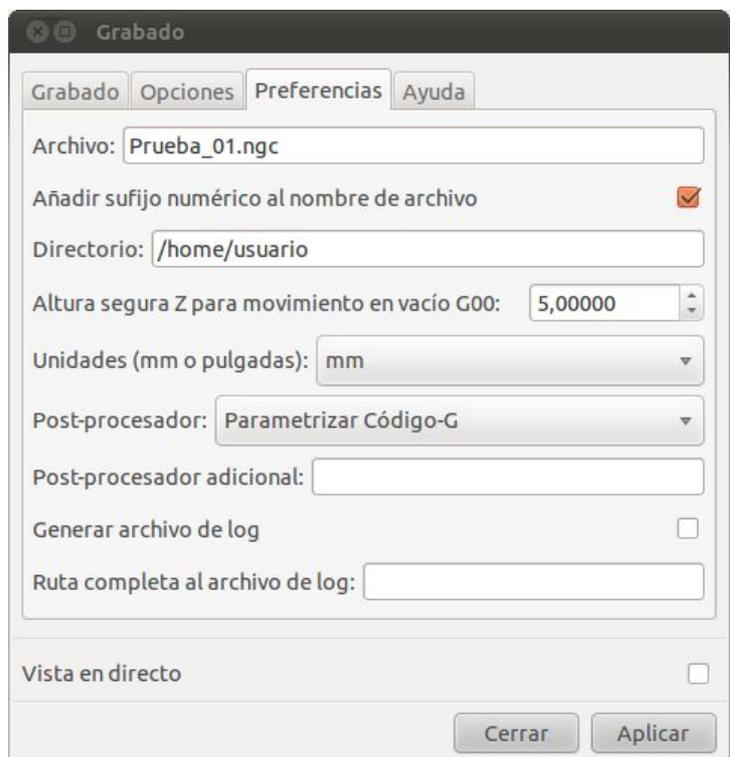
En el campo **Archivo** introduciremos el nombre que queremos que tenga el archivo de Gcode que se generará.

Si marcamos **Añadir sufijo numérico al nombre de archivo**, el programa irá añadiendo un número consecutivo al final del nombre para evitar sobrescrituras. Debemos asegurarnos que el **Directorio** que elijamos exista, y que además tengamos permiso de escritura en él. Con el valor que introducimos en **Altura segura Z para movimiento en vacío G00**, definimos la altura del plano en el que se moverá la fresa cuando no esté mecanizando.

En **Unidades** especificamos las unidades en las que se generará el Gcode en milímetros o pulgadas.

En el cuadro desplegable de **Post-procesador** podemos elegir entre diversas opciones:

- **Ninguno**
- **Parametrizar códigoG**: Si elegimos



esta opción obtendremos código similar a esto:

```
G03 X[428*#5+#8] Y[-590*#5+#9] Z[-20*#7+#10] I[0*#5] J[45*#5]
```

Y en el encabezamiento tendremos algo similar a:

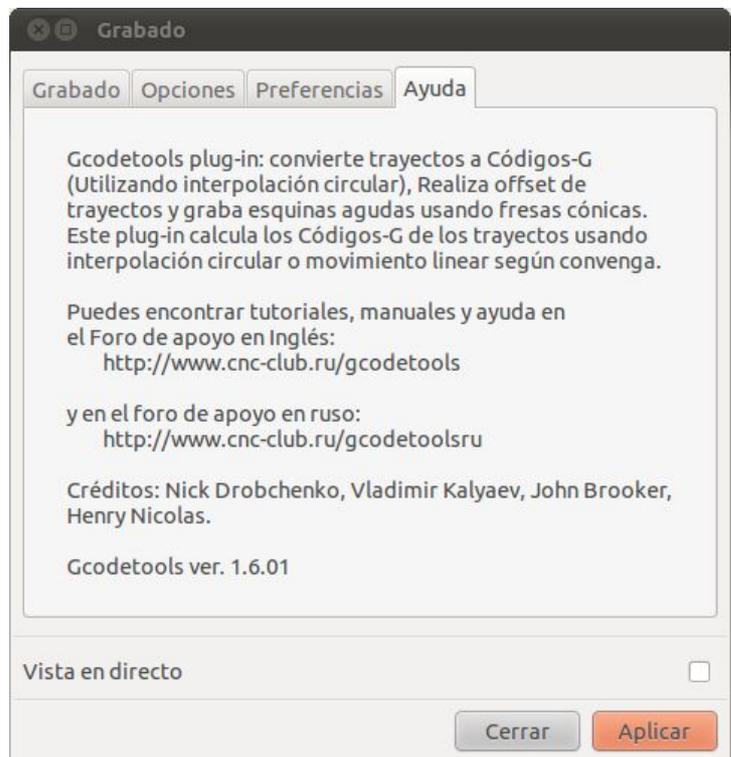
```
#4 = 700.000000 (Feed)
#5 = 1.000000 (Scale xy)
#7 = 1.000000 (Scale z)
#8 = -300.000000 (Offset x)
#9 = 300.000000 (Offset y)
#10 = 0.000000 (Offset z)
#11 = 10.000000 (Safe distance)
```

De forma que si modificamos estos parámetros, podemos modificar el programa Gcode fácilmente.

- **Invertir Eje Y y Parametrizar código**, realiza lo explicando en el punto anterior, y además genera el código de forma que realiza la simetría con respecto al eje Y.
- **Redondear todos los valores a 4 dígitos**.

Ayuda

Simplemente muestra un texto informativo del programa y los enlaces de los foros de ayuda y descarga.



Veremos ahora una descripción de las lengüetas "Activas"

Puntos de Orientación

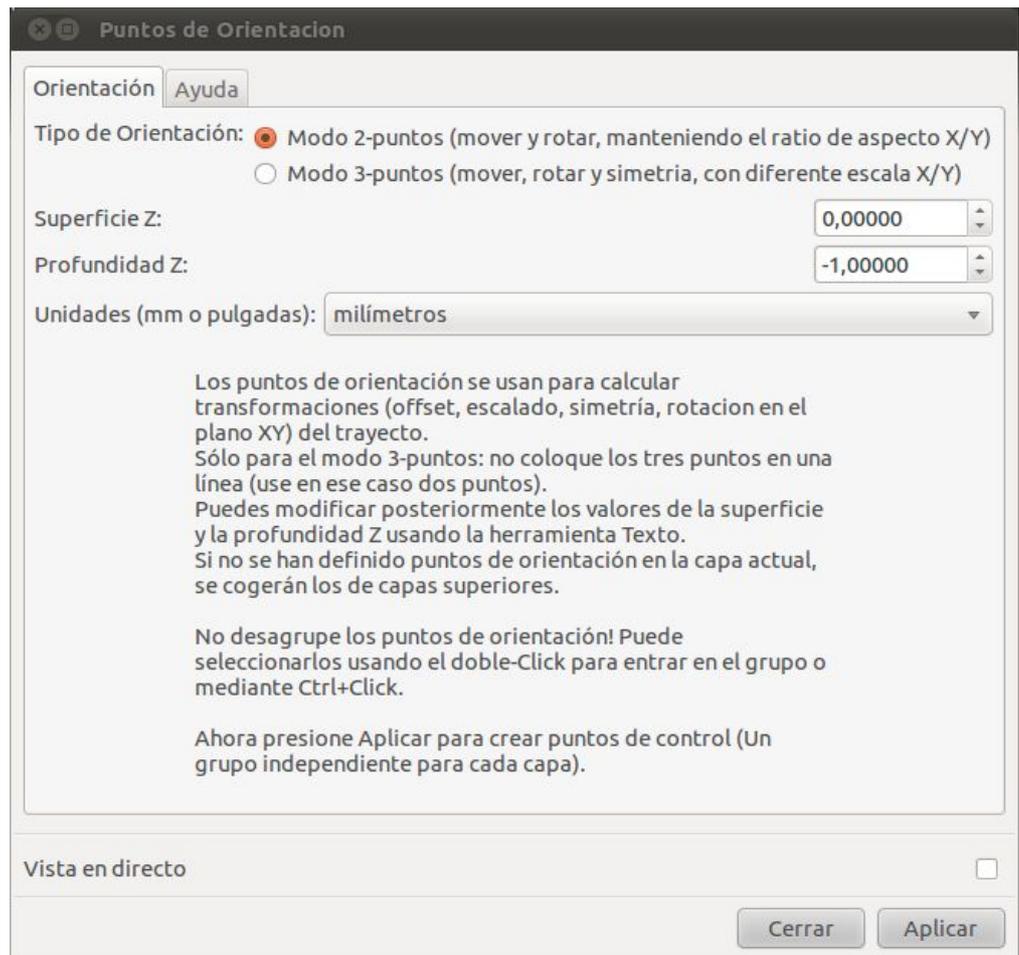
Gcodetools necesita para sus cálculos definir por lo menos dos puntos como referencia.

Al apretar el botón de **Aplicar**, creará en el documento inkscape dos puntos situados en la parte inferior de la hoja.

El vértice de la hoja será el origen de coordenadas para el programa Gcode.

Puedes leer las recomendaciones contenidas en la propia lengüeta.

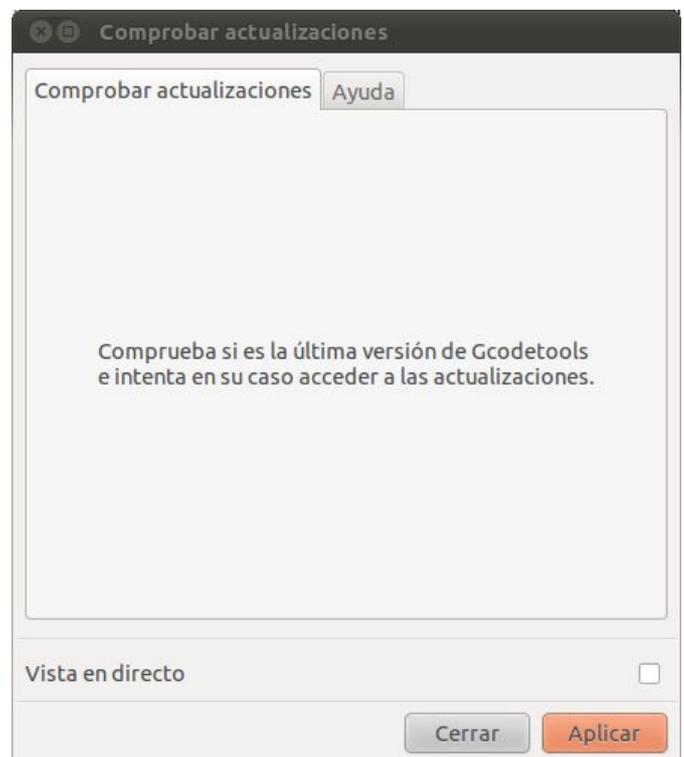
NOTA: Si intentamos realizar alguna operación con Gcodetools sin haber definido previamente unos puntos de Orientación, el programa intentará crearlos.



Comprobar actualizaciones

Gcodetools puede comprobar si estás utilizando la última versión del programa.

Al apretar el botón **Aplicar** buscará en la web de descargas si existe una versión más actualizada del programa.



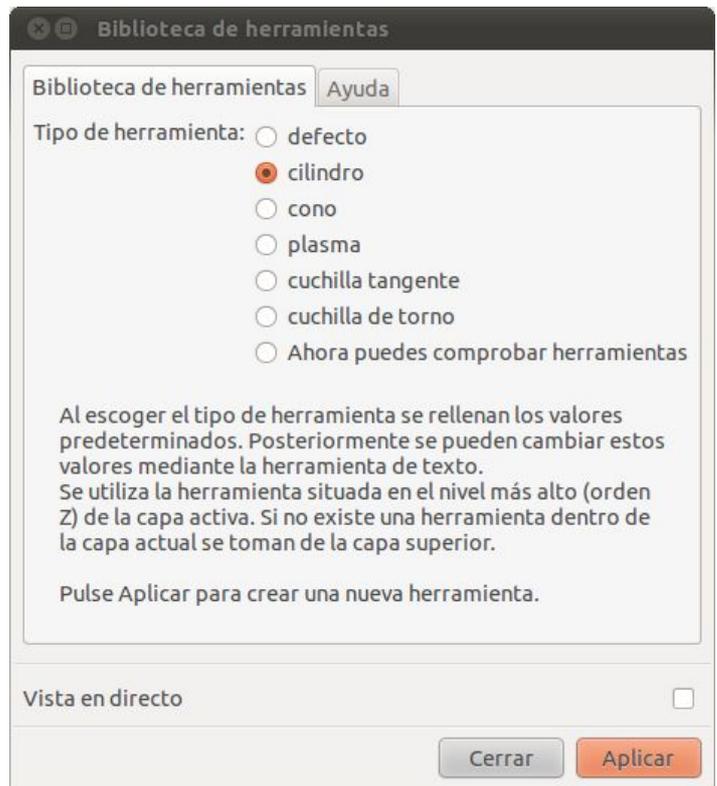
Biblioteca de Herramientas

Mediante esta función asociaremos una herramienta a nuestro mecanizado.

Al apretar sobre el botón **Aplicar**, el programa creará una herramienta, en la capa activa de nuestro documento inkscape, con los valores por defecto.

Con la herramienta de texto  de inkscape podemos modificar los valores (La descripción no) de diámetro, Velocidad de avance, ángulo de penetración, velocidad de penetración, ...

name	Cylindrical cutter
id	Cylindrical cutter 0001
diameter	10
feed	400
penetration angle	90
penetration feed	100
depth step	1
tool change gcode	



Estos valores son los que tomará el programa cuando generemos el archivo de Gcode.

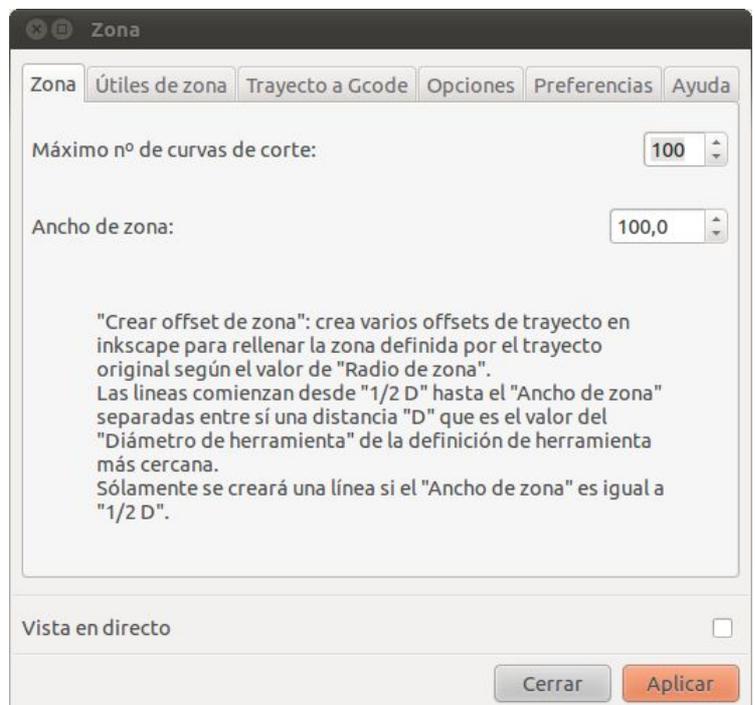
NOTA: Si ordenamos ejecutar a Gcodetools algún comando que necesite una herramienta, y todavía no la hemos definido, el programa creará una por defecto automáticamente.

Zona

El valor que se introduzca en Máximo número de curvas de corte, nos definirá la cantidad de curvas de corte que se crearán. Este límite es para evitar que el programa entre en un bucle infinito.

El Ancho de zona es la distancia máxima que tendrá la zona a mecanizar. Si es negativo considerará la zona exterior. Existen varios métodos para saber el valor de esta variable:

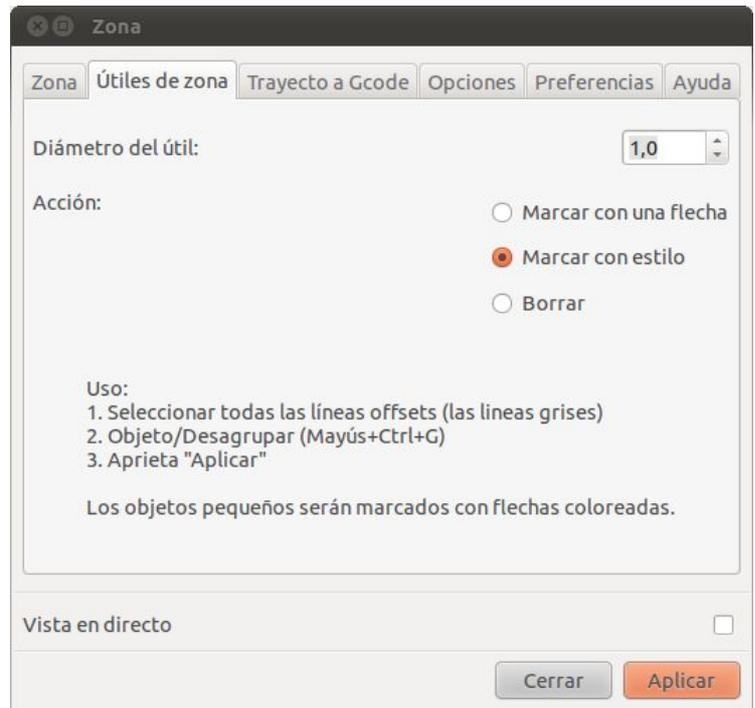
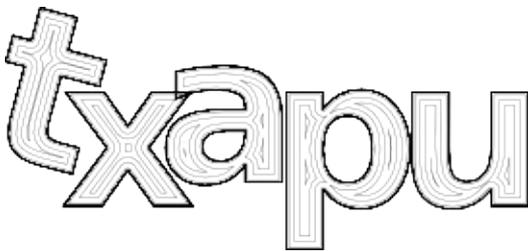
- La forma más fácil es: encontrar la parte más ancha de la trayectoria y medirla a ojo ;-)
- La segunda forma es: encontrar la parte más ancha de la trayectoria y medirla con la herramienta Bizier. Al dibujar una línea con la herramienta Bezier, en la barra de estado veremos su longitud.
- Y por último, la manera más dura es: convertir el trayecto en Desvío Dinámico (Ctrl + J), a



continuación, presione F2 y mover el punto de control para disminuir al máximo el trayecto. A continuación, pulse Ctrl+SHIFT+X para abrir el editor de XML y buscar el parámetro `inkscape:radius`, este valor es el valor exacto que necesitaremos.

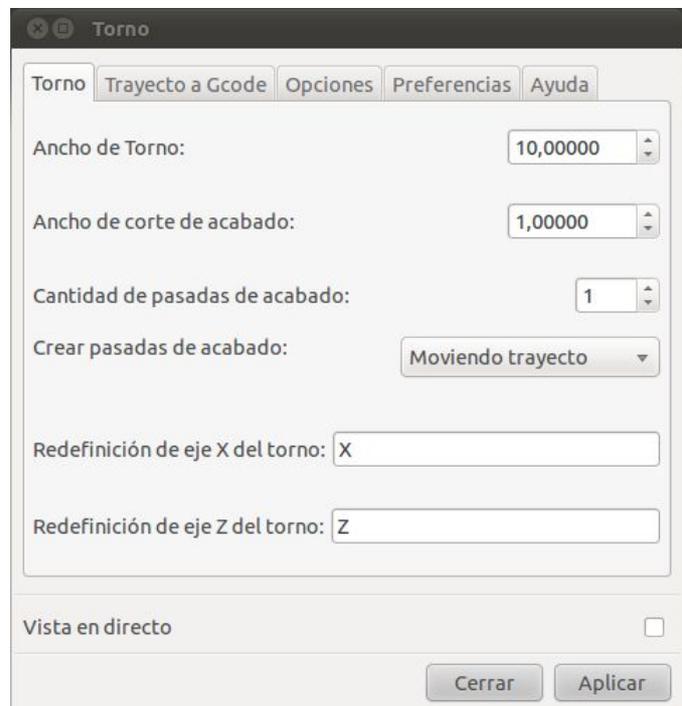
Como esta herramienta genera curvas defectuosas, tenemos otra lengüeta, la de útiles de Zona, que nos ayudará a descubrir las pequeñas líneas creadas por error (Al menos eso creo, ya que no he conseguido que me funcionara)

Al apretar el botón **Aplicar**, el programa creará en inkscape los trayectos desplazados, para que luego los exportemos con **"Trayecto a Gcode"**



Torno

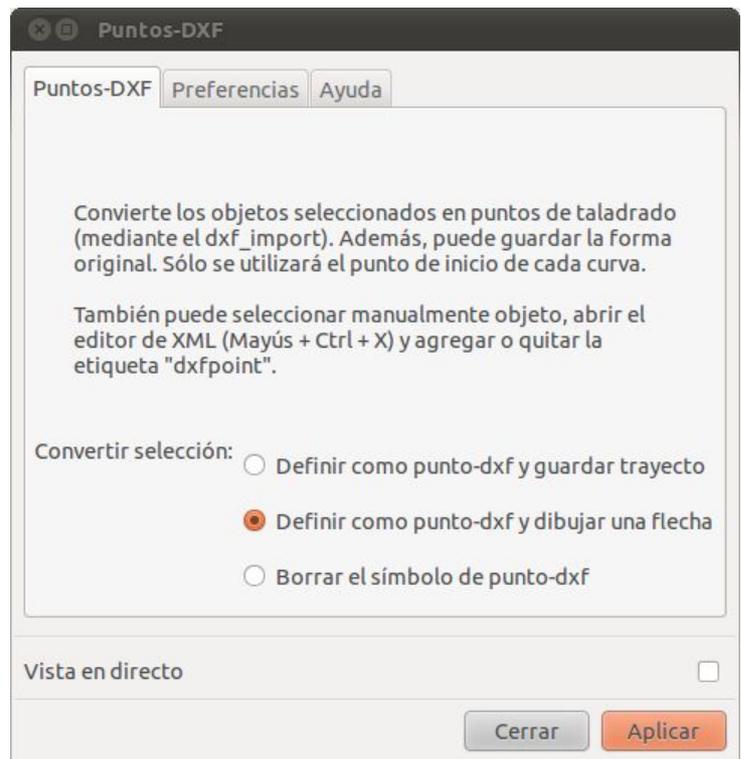
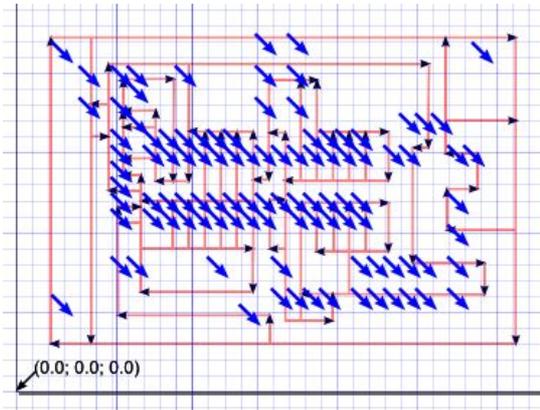
No sé que hace esta lengüeta.
No tengo torno :-)



Puntos-DXF

Cómo se indica en la lengüeta, al apretar el botón **Aplicar**, asociaremos a los objetos seleccionados una etiqueta "dxfpoint" (si seleccionamos alguno de los dos primeros items).

Si seleccionamos la segunda opción , en inkscape veremos los puntos con una flecha.



En el ejemplo vemos un detalle del archivo inkscape del mecanizado de una placa PCB para paperduino (con trayectos en rojo y puntos de taladrado en azul).

NOTA: Una vez creado un Punto-DXF puedes en inkscape copiarlo y pegarlo cada vez que quieras un taladrado.

Posteriormente , cuando generemos el Gcode, el programa generará un proceso de taladrado en cada punto-DXF.

```
...
(drilling dxfpoint)
G00 Z5.000000                                (sube a la cota de seguridad)
G00 X75.440241 Y52.539318                    (se mueve al próximo agujero)
G01 Z-1.000000 F100.000000                  (taladra a velocidad 100)
G04 P0.200000                               (espera un poco)
G00 Z5.000000                                (sube a la cota de seguridad)
(drilling dxfpoint)
G00 Z5.000000
G00 X74.094210 Y36.072453
G01 Z-1.000000 F100.000000
G04 P0.200000
G00 Z5.000000
...
```

Grabado

Mediante esta función podemos mecanizar (grabar) aprovechando la forma de herramienta, es decir si utilizamos una fresa cónica como en la imagen de la derecha, podemos modificar la anchura de mecanizado controlando la altura Z de mecanizado. De esta forma podemos mecanizar ángulos agudos.

El programa **Codetools** lo que hace es modificar la trayectoria de mecanizado para compensar las diferencias de corte al modificar la altura. Un ejemplo lo puedes ver en la imagen de la derecha donde comparamos los diferentes trayectos del proceso de grabado y el del proceso de exportación a Gcode.

Utilizando este sistema, podemos realizar mecanizados similares al siguiente:



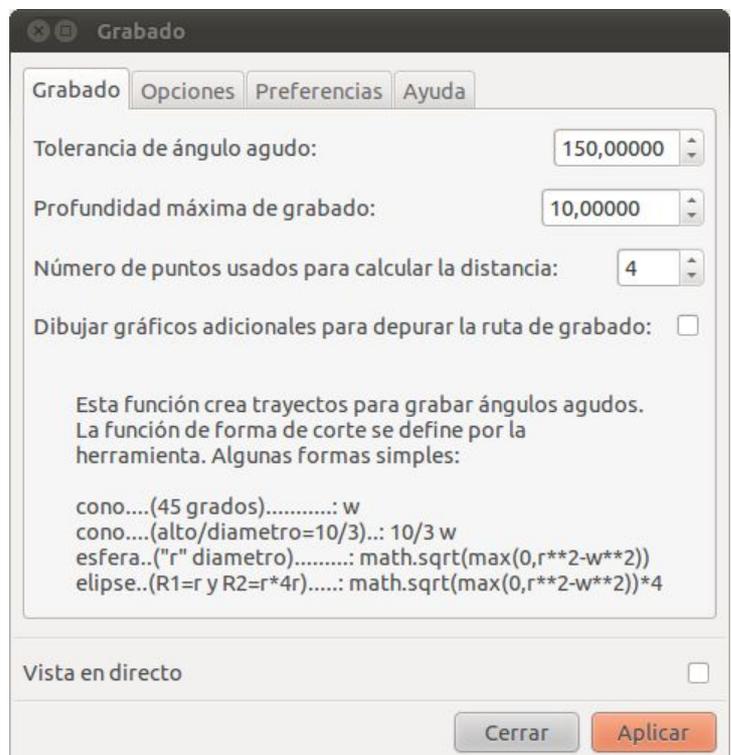
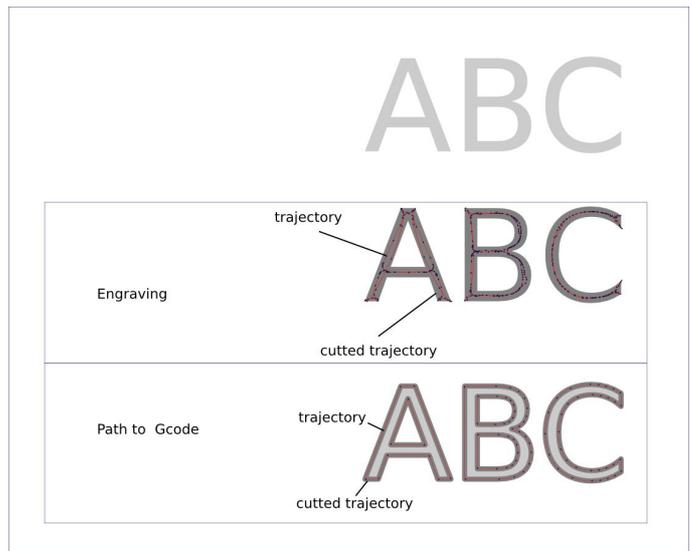
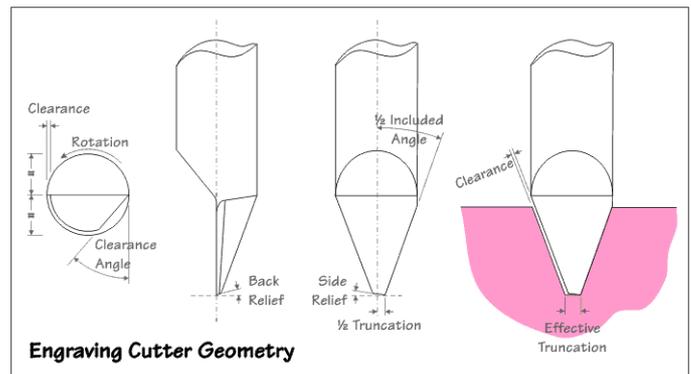
El valor de la **Tolerancia de ángulo agudo** indica el valor en grados a partir del cual el programa lo considerará como vértice agudo.

Si el ángulo del vértice es muy agudo, la trayectoria de corte llegará a tocar el vértice.

Si el desplazamiento en Z de la trayectoria supera la **Profundidad máxima de grabado**, este trayecto no será utilizado para el corte. Este valor debe ser igual al diámetro de la fresa cilíndrica (que se utiliza para cortar la zona interior) o debe ser grande si se desea cortar camino con una fresa cónica.

El **Número de puntos usados para calcular la distancia** define el número de puntos de referencia usados para el cálculo del desplazamiento de la trayectoria. El tiempo de cálculo depende directamente de este parámetro. Debería ser entre 4 y 10.

Si activamos **Dibujar gráficos adicionales para depurar la ruta del grabado**, el programa dibujará



graficos auxiliares (normales, puntos de desplazamiento, circulos, ...) que nos pueden ayudar a encontrar errores, o al menos no ayudará a encontrar una manera de cambiar la ruta de mecanizado para eliminar los errores.

Hay que tener en cuenta que el tiempo de cálculo depende del cuadrado del número de los puntos de control de ruta. Si se va a utilizar esta función en un trayecto con 200 puntos de control o más, tardará bastante en hacer todo el trabajo. Si los caminos no se cruzan, es mejor separarlos. Un usuario (Lirón) a comentado que cuando trató de calcular la ruta de 2.000 puntos el programa tardó 15 minutos para hacer los cálculos.

Trayecto a Gcode

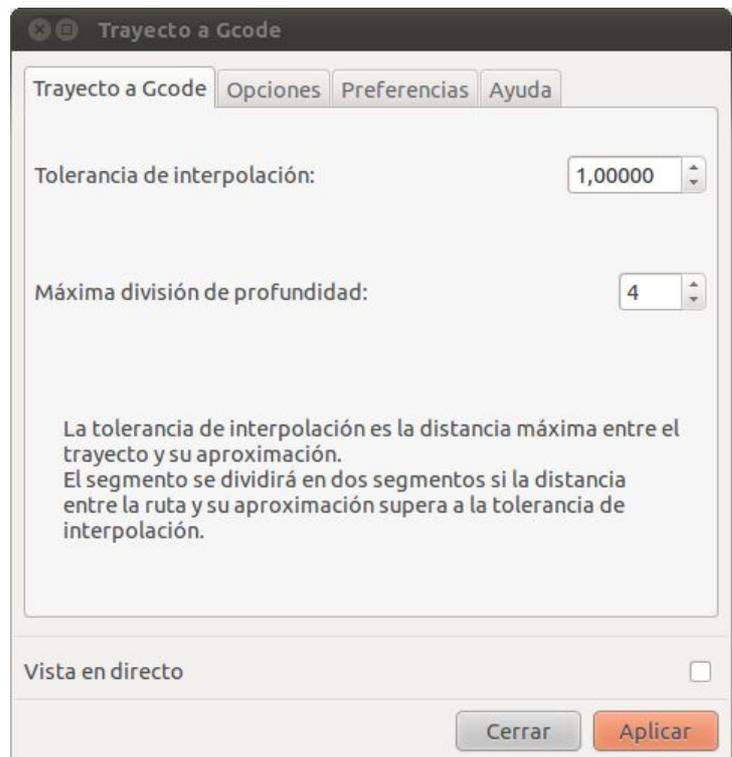
Como se indica en la lengüeta, la **Tolerancia de interpolación** es la distancia máxima entre el trayecto y su aproximación. Es decir el error máximo que admitimos. Cada vez que se aproxima el segmento de trayecto, se calcula esta distancia, se dicha distancia es superior, el segmento se divide en dos segmentos, y cada uno de ellos se aproxima por separado.

La **Máxima división de profundidad** define cuantas veces puede ser dividido un segmento.

Al pulsar el botón **Aplicar** generaremos el programa Gcode de los trayectos que tengamos seleccionado.

***Nota:** es importante repasar los valores de las lengüetas Opciones y preferencias sobre todo el campo directorio en el que seleccionaremos el directorio donde queremos guardar el archivo de Gcode que se generará, ya que por defecto el directorio es inválido:*

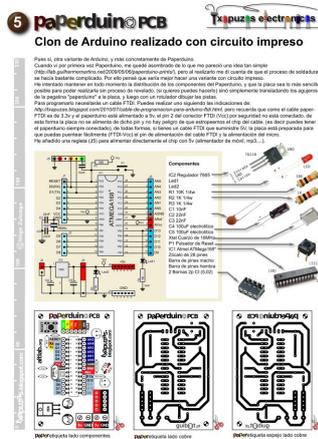
*El Linux `/home/` => `/home/[TU_NOMBRE_DE_USUARIO]...`
En Windows `/home/` => `c:\...`*



Ejemplo de uso: Realización de paperduino-PCB (txapuzas.blogspot.com)

Primeramente descargamos el archivo de la página web de txapuzas:

<http://txapuzas.blogspot.com/2010/07/paperduino-pcb.html>



1 Lo descomprimos y abrimos el documento Paperduino-PCB_01.svg en inkscape

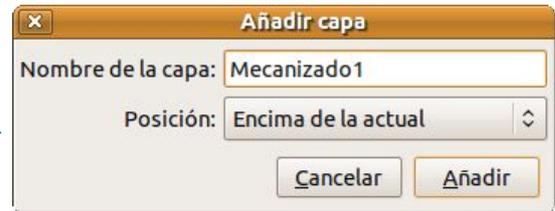
2 Borramos todo salvo la parte de circuito impreso, y lo aproximamos a la parte inferior izquierda de la hoja (origen de coordenadas por defecto).



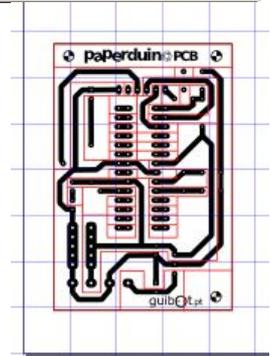
3 Ahora bloqueamos esta capa para no modificarla accidentalmente



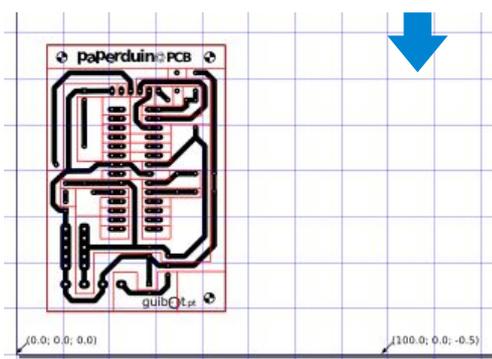
4 Creamos una capa nueva con Capa/Añadir Capa ó (Mayús+Ctrl+N) y le damos el nombre de Mecanizado1



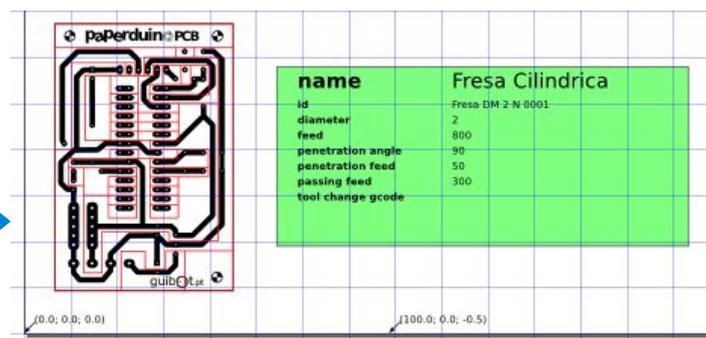
5 En esta nueva capa, con la herramienta de Bezier (curva) vamos dibujando las líneas de separación de pistas (yo he dibujado todo con líneas rectas por sencillez)



6 Seleccionamos **Extensiones/Gcodetools/Puntos de Orientación**. En el cuadro de diálogo definimos a que altura está la cara superior de la pieza (normalmente a 0mm), y cuánto queremos que profundice el mecanizado (entre -0,5 y -1 mm), y Gcodetools nos definirá los puntos de control.



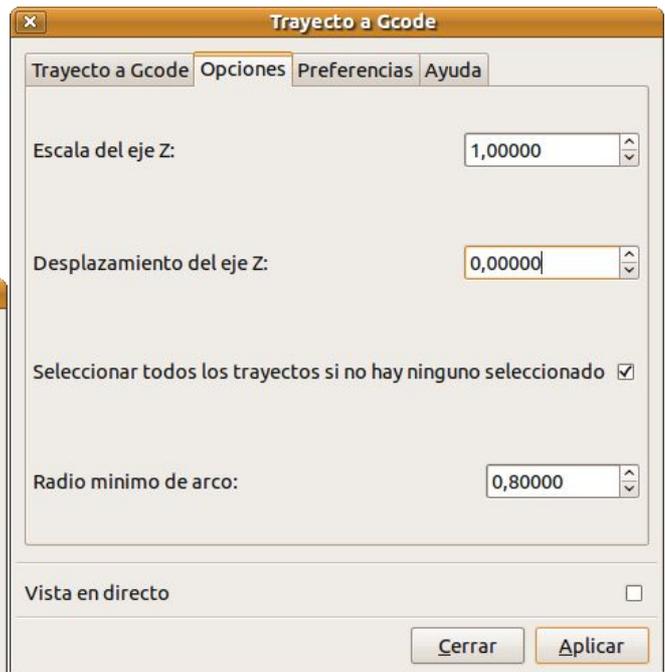
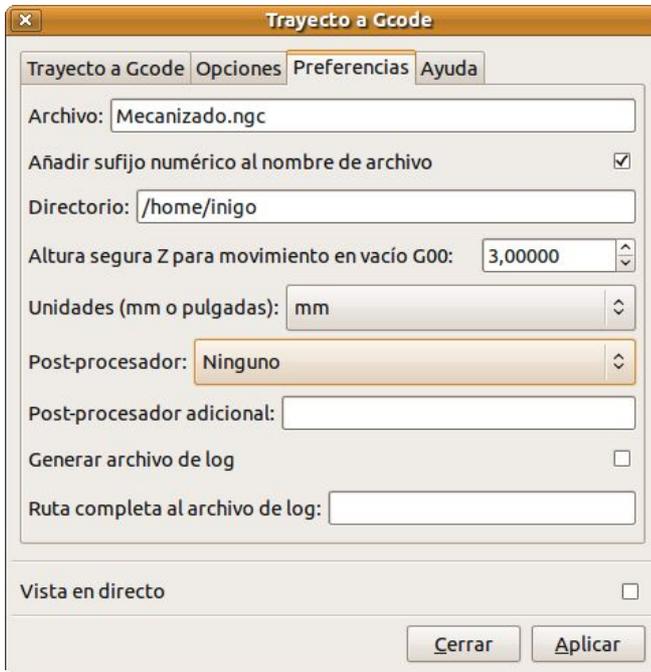
7 Seleccionamos **Extensiones /Gcodetools/Biblioteca de Herramientas**. Seleccionamos la forma de nuestra fresa, y al apretar el botón de aplicar, nos creará un cuadro con los valores por defecto de la herramienta.



Con la herramienta de texto modificaremos los campos de Diámetro de fresa, Velocidad en vacío, Ángulo de penetración, Avance ... de acuerdo a la fresa que utilizemos, el material a mecanizar, es decir, lo adecuamos a nuestras necesidades.

8 Seleccionamos los trayectos que queremos mecanizar y apretamos en el menú **Extensiones/Gcodetools/Trayecto a Gcode**

9 Comprobamos que los valores de la lengüeta **Opciones** son correctos.



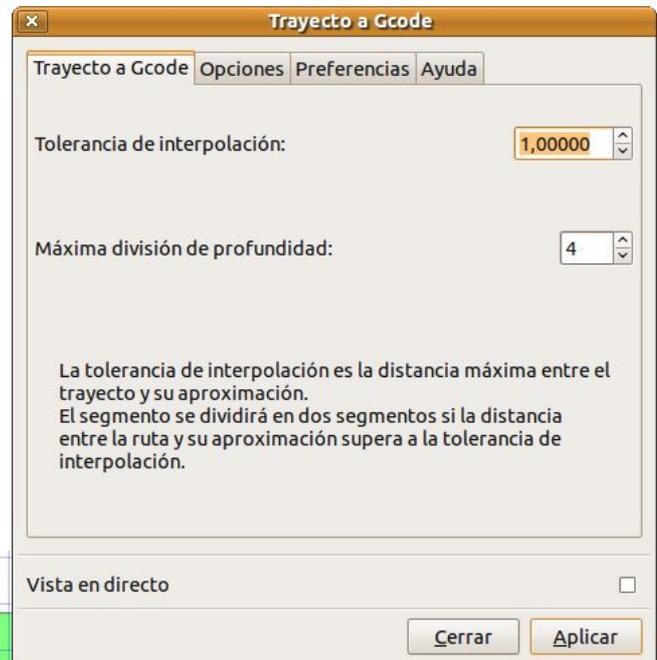
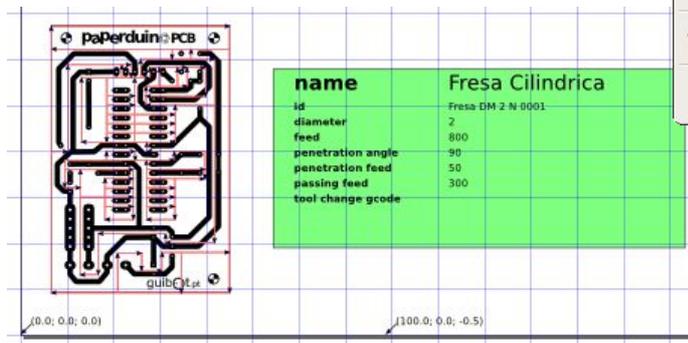
10 Comprobamos también que los valores de la lengüeta **Preferencias** son correctos.

11 Y por fin vamos a la lengüeta **Trayecto a Gcode**. Comprobamos también los valores

(En la imagen, El valor de la tolerancia de interpolación es "un poco grande")

Al apretar el botón **Aplicar**, veremos que nos representa en el archivo inkscape la ruta de mecanizado indicando el sentido de movimiento en cada trayecto.

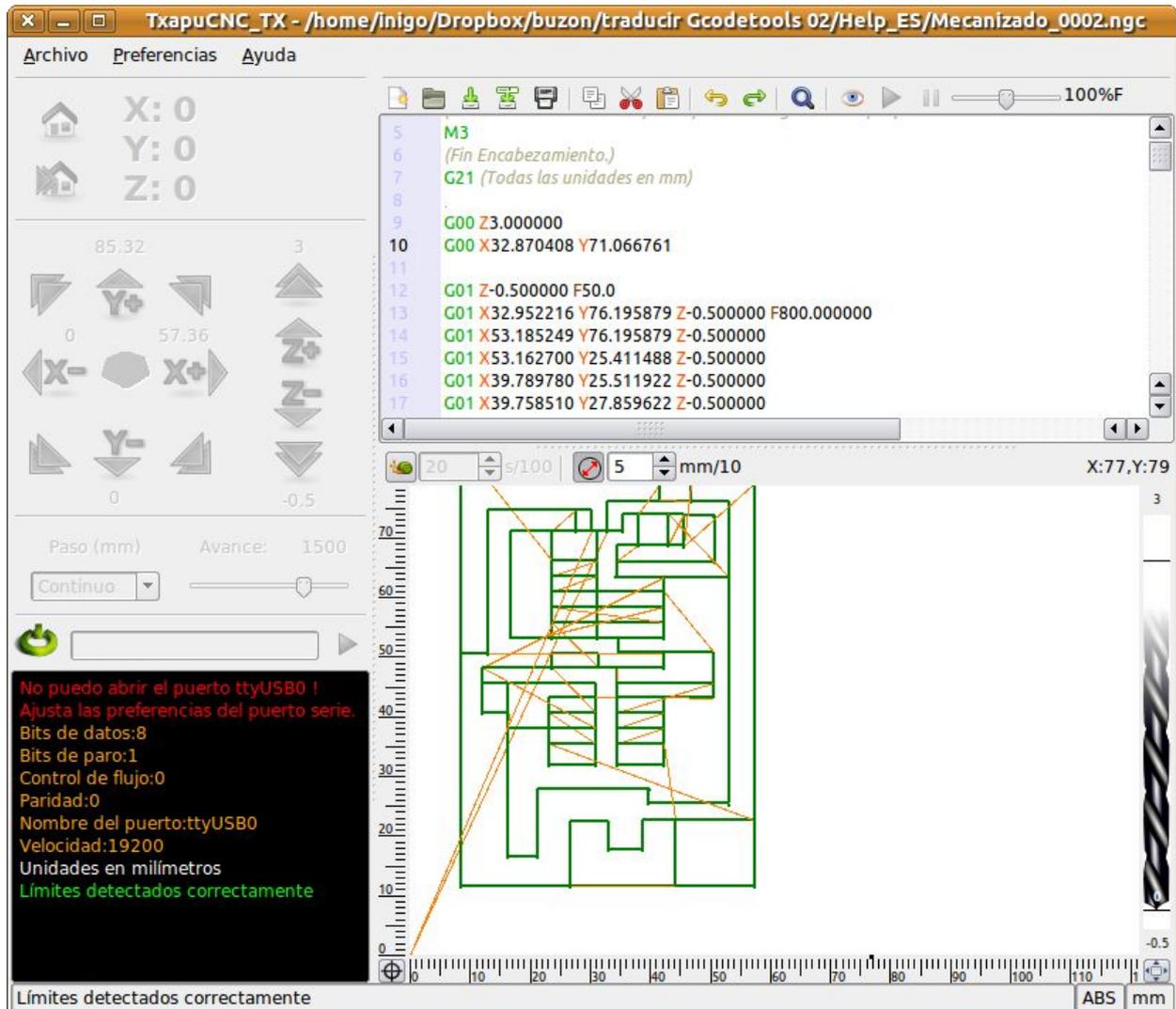
Pero lo más importante es que nos ha generado



en el directorio que le hemos indicado el archivo con el Gcode listo para mecanizar.

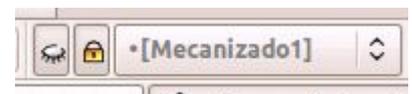
El archivo generado lo podemos abrir con txapuCNC_TX.gambas (lo puedes descargar en <http://txapuzas.blogspot.com/2009/12/txapu-cnc-software.html>)

y ahí comprobarlo y editarlo



Ahora vamos a hacer las operaciones de taladrado.

12 Como antes, en inkscape primero **bloqueamos la capa anterior** (Mecanizado 1) para no modificarla por error. (apretando el candado del panel inferior) y también podemos ocultar esta capa para no moleste.



13 Creamos una capa nueva que denominaremos Taladrado1

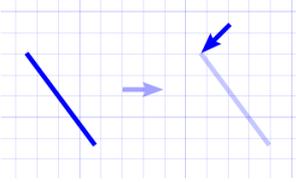
Apertando Capa/Añadir Capa ó (Mayús+Ctrl+N)



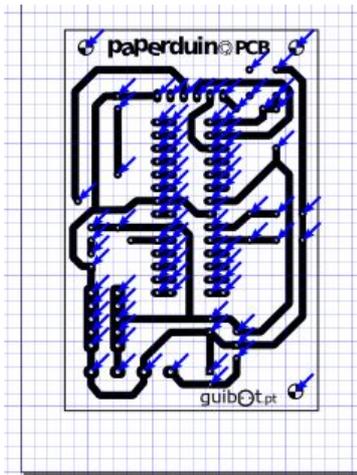
14 Dibujamos una línea, **la seleccionamos** y apretamos **Extensiones/Gcodetools/Puntos-DXF** y seleccionamos la lengüeta **Puntos-DXF**.

Elegimos la segunda opción (**Definir como punto-dxf y dibujar una flecha**) y apretamos el botón **Aplicar**.

Y entonces mágicamente la línea desaparece y en el punto inicial de la línea aparece una flecha, éste es el símbolo de punto-DXF (internamente en la descripción del objeto tiene la etiqueta dxfpoint).

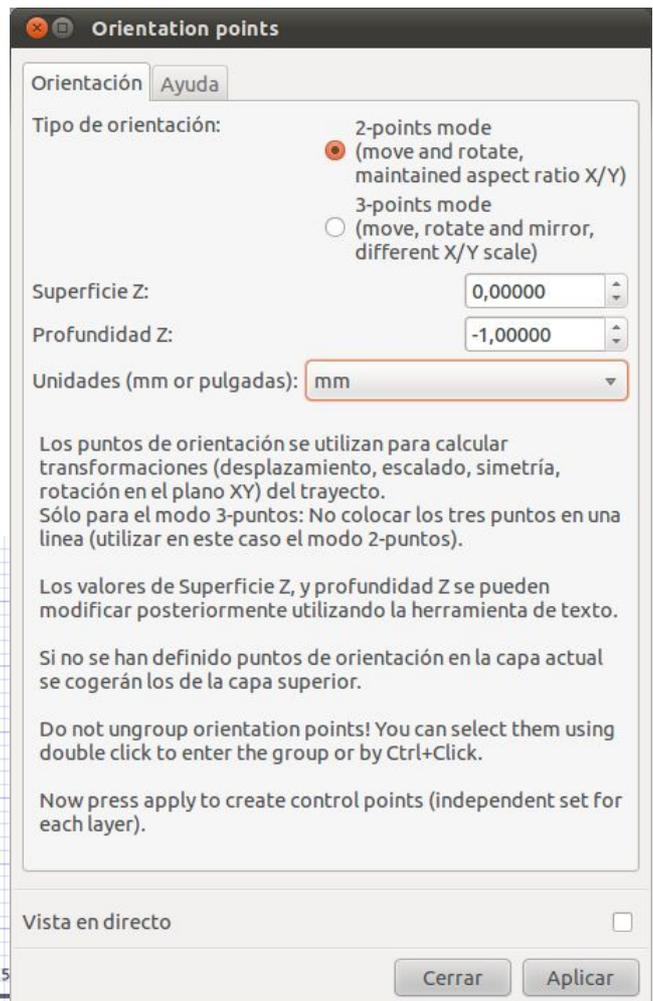
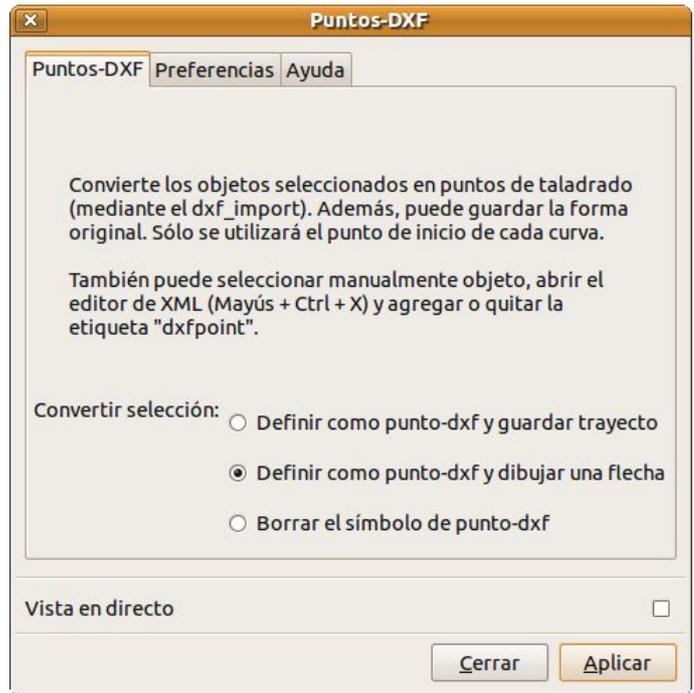
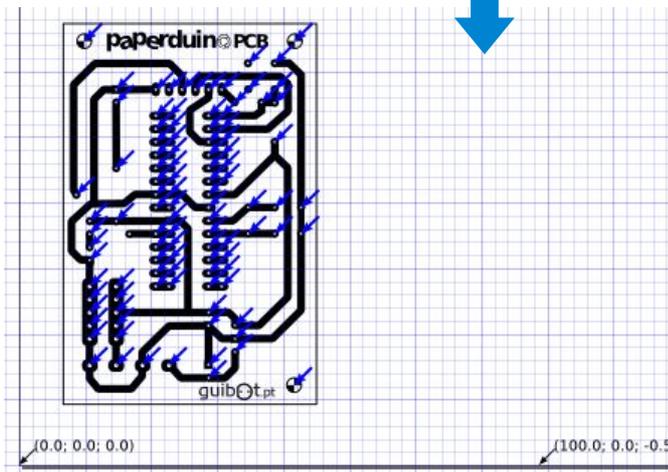


15 Esta flecha la movemos hasta la posición donde queremos hacer un agujero, la **copiamos** (Ctrl+C) y **pegamos** (Ctrl+V) tantas veces como agujeros deseemos.



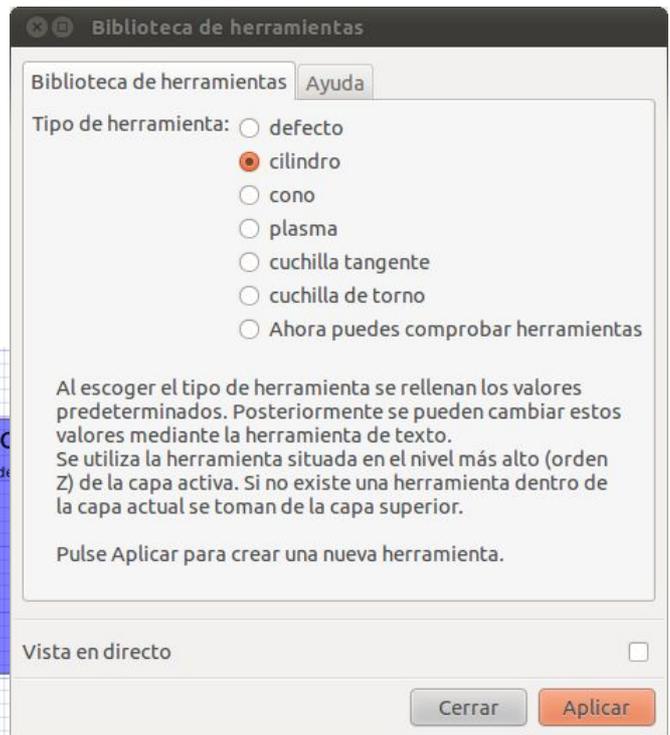
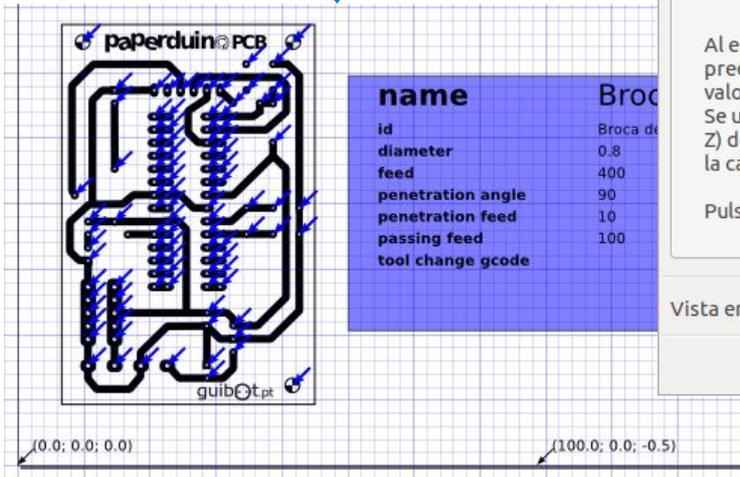
16 Definimos los **Puntos de Orientación** seleccionando en el menú de Inkscape **Extensiones/Gcodetools/Puntos de Orientación**.

Apretamos el botón de **Aplicar** y nos creará los puntos en el documento inkscape:



17 Aplicamos una herramienta a la capa activa seleccionando en el menú: **Extensiones/Gcodetools/Biblioteca de Herramientas**. Nos aparecerá un diálogo donde elegiremos la forma de nuestra fresa. 

Al apretar el botón de **Aplicar**, nos creará un cuadro con los valores por defecto de la herramienta.



18 Seleccionamos todas las flechas y apretamos en el menú **Extensiones/Gcodetools/Trayecto a Gcode**

9 Comprobamos que los valores de la lengüeta **Opciones** son correctos. 



10 Comprobamos también que los valores de la lengüeta **Preferencias** son correctos. (sobre todo el nombre del archivo) 

11 Seleccionamos la lengüeta **Trayecto a Gcode**

y apretamos el botón **Aplicar**.

En ese momento, si todo ha salido bien ;-)) el programa nos generará un Archivo con código G con las instrucciones de taladrado.

12 El archivo generado lo podemos abrir con **txapuCNC_TX** (lo puedes descargar [aquí](#))

Donde podemos editarlo y modificarlo según nuestras necesidades

