

# CNC Router User Manual

Nathan Paige

January 16, 2017

## 1 Overview

### 1.1 Mechanical

### 1.2 Electronics

The router is fairly similar to a 3D printer electronically: it has an Arduino Uno for the brain, which sends signals to stepper drivers to control the stepper motors to move the axes. There is additionally a variable speed spindle motor (0-100VDC, 600W) with its own separate power supply. The main power input to the router is 220VAC. Limit switches are used for endstops.

### 1.3 Software

There are three components to the software for the router: the firmware, which runs on the onboard Arduino, the host, which runs on a computer connected to the router, and the wizards, which facilitate generating command sequences for common operations.

The firmware does all of the low-level control. It communicates with the host computer, from which it receives a stream of commands, which it interprets and executes. It figures out when to make the motors step to achieve a given movement, keeps track of its position, and possibly applies corrections for rotation and skew. Firmware can be found at <https://github.com/albinoBandicoot/router-firmware>.

The host software feeds commands to the router. It can be used to manually move the router around as well as run command sequences saved in files. The router can also be queried for certain information, which the host will then display. The host is written in C++ using OpenGL, and is available at <https://github.com/albinoBandicoot/router-host>.

Typing up command sequences manually is tedious and error-prone, so the wizards automate the process for common operations. This also allows editing a sequence of operations at a higher level (changing parameters to a sequence of wizard invocations rather than mucking with individual commands). The wizards are written in Java (and you can write your own, too!) and are available at <https://github.com/albinoBandicoot/router-wizards>.

## 2 Milling Fundamentals

### 2.1 Safety

Mills and routers are dangerous. This one is more dangerous, since I designed and built it. Don't trust me too much; always double-check things; if you're uncertain, talk to someone (you can email me at npaige@oberlin.edu). You *must* understand this thing before you can use it.

Wear safety glasses. Tie back long hair. Make sure it's not going to start unexpectedly. Keep your hands away from the spinning sharp thing.

Make sure the tool and the workpiece are held securely.

Check your feeds and speeds.

If it sounds bad, it probably is bad. Seriously though – sound is one of the most important diagnostic tools in milling.

### 2.2 Tools and Collets

There are three main types of tool that are used in a router: edgefinders, drill bits, and endmills. Edgefinders are used for locating features on parts (edges, centers of holes, that sort of thing) to establish where the router is. Drill bits are used exclusively for drilling holes. Endmills are the real workhorse of the router.

There are many types of endmills. The biggest difference between an endmill and a drill bit is that the endmill has cutting edges running up its sides (both drill bits and endmills have flutes, but on drill bits these are not sharpened). A spinning endmill can be pushed sideways into some material, and it will carve it away. A drill bit would likely break in the same situation.

Most endmills have a flat bottom, with cutting edges both on the bottom and spiraling up the sides. This enables them to both plunge into material and cut sideways. Some endmills have other shapes: there are tapered endmills, ball-nose endmills, endmills with rounded corners, dovetail cutters, T-slot cutters, etc.

To attach the tool to the spindle motor, a collet is used (this router uses ER-11 collets). The collet has a central hole, surrounded by metal that has slits in it. This creates a ring of slightly bendable metal fingers. For this style of collet, the tool is inserted into the hole, and then a nut around the outside is tightened. This compresses the fingers, which press against the tool. The friction (surprisingly) is sufficient to hold the tool. If a collet is overtightened, the metal fingers can be permanently deformed, impairing its ability to hold tools.

A collet is only capable of holding tools with a specific shank diameter. Endmills come with shanks in a few standard sizes (the cutting part is often a slightly different size than the shank); however, drill bits usually have a shank diameter that's the same as the drill diameter. This makes it difficult to use drill bits (except for sizes that match the collet size) in the router.

However, it is possible to get an adjustable chuck that can go in a collet, and then put drill bits in the chuck.

## **2.3 Types of Operation**

There are many different types of operations that can be done on a router. However, before anything can be done, the location of a reference point on the part must be determined.

### **2.3.1 Edgefinding**

Edgefinding is the process of accurately determining the position of an edge of a part (or the jaw of the vice). Positions will need to be established along all three axes.

The Z axis is somewhat different than the X and Y.

### **2.3.2 Drilling**

The most basic operation is drilling holes. This can be accomplished with either a drill bit or an endmill. Drill bits are pointed and generally cut better for drilling than endmills (which are generally flat-bottomed) do. If using an endmill to drill, make sure that it is ‘center-cutting’ – that is, the cutting edges extend all the way to the center of the tool.

Drilling, if done in one continuous motion, can produce very long curly streamers of material. This is a good indication that the spindle speed and feedrate are good. However, these streamers are sharp and spinning rapidly. In deep holes, chips are harder to clear and can start plugging up the flutes of the tool. If this happens, the heat can melt them onto the tool, significantly reducing performance and creating even more heat. Peck-drilling is a way to counter this: the tool repeatedly drills down for a bit and then retracts briefly before resuming. This helps to both break long chips and lift compacted chips out of deep holes.

Drilling large diameter holes is best done in multiple passes, starting with a small drill bit or endmill and progressing to larger sizes. Larger drill bits have an area at the center which is not particularly sharp/good at cutting. Also, the lower speeds needed for the larger bits mean areas near the center are receiving very little cutting action. It is best to have a small hole that the tip can sink into, thus centering the drill. Larger bits also require more torque; if they are cutting across their entire diameter this is increased even further. This router’s spindle motor is not all that powerful, particularly at low speed. Pocketing is a better approach for large holes.

Drill bits are rather long and thin, so bending can be a problem. The drill bit tip can skate around on the surface of the part for a while before it actually catches and starts cutting. This can significantly reduce the accuracy of the hole location. When accuracy is important (which it usually

is – otherwise you could just be using a hand drill), it is a good idea to use a center drill first.

The large shank diameter and very short length of the cutting area means that there will be essentially no bending, so the hole location will be accurate. Center drills are not typically used to actually drill the hole; rather, they are used to make an accurately positioned divet in the surface. Then, a drill bit can be used, and the point will center itself in the divet, causing the hole to be drilled in the correct place.

Endmills would not benefit from a center drill, as there the tool has no point to guide it into the divet. However, endmills are usually rigid enough to just drill on their own.

### **2.3.3 Facing**

When stock has been cut roughly with a saw or something similar, and a precisely flat and square edge is desired, along with a specific length, an endmill can be used to remove material along the side of the part. This is called facing, because it produces a pristine, precise face.

The easiest setup for facing is to have the edge of a part protruding slightly from the vice. The mill can then make passes over the exposed edge until the desired length is reached.

### **2.3.4 Surfacing**

Surfacing is like facing, but along the Z axis. The Z axis is lowered, the router makes a pass over the surface of the part, and possibly repeats. This is good for making parts thinner, for making a precise flat surface, or making a flat area on a cylindrical part.

### **2.3.5 Pocketing**

Pocketing refers to making cutouts, either all the way through or only part-way. It is essentially a localized surfacing operation. It has many applications, including making slots, making large diameter holes (or really, any hole larger than the endmill: this is great for making holes for which you don't have the right drill bit), making grooves, creating raised or lowered areas, etc.

### **2.3.6 Thread Cutting**

A special tool can be used to cut threads with a router. The tool looks like a stick with a small triangular piece of metal sticking out the side near the tip. A cylindrical piece of material is mounted vertically, and then the router can move in a helix around the outside of the piece while spinning the tool. The triangular bit will cut out a thread. This can also be used

for internal threads in a hole. These tools are rather expensive, especially considering that to make different thread sizes, different tools are needed.

## **2.4 Work Holding**

Keeping the part held securely is critical for safety, preventing tool damage, and accuracy. If the part becomes dislodged while in contact with the spinning tool, it can be ejected from the router at high speed (and will probably be ruined in the process). If the part moves while cutting, subsequent operations will not happen in the right place. Finally, long overhangs or unsupported sections will bend and vibrate, decreasing accuracy and causing chatter, which decreases tool life, produces poor surface finish, and makes awful noises.

### **2.4.1 Vice and Parallels**

For small parts, the best solution is typically to clamp the part in a vice. The vice is bolted into the bed of the router, and precisely aligned with the router axes (the jaws are parallel to the X axis). The jaws of the vice are smooth and precisely flat and parallel. A part with a pair of flat, parallel faces opposite each other can be clamped between the jaws.

However, something must be done to control the Y-axis rotation of the part. For this, parallels are used. Parallels are precision-ground rectangular blocks of steel. They typically come in sets of several pairs with various heights. An appropriate pair is selected depending on the shape of the part, and the parallels are placed up against the jaws of the vice. The part can then be placed on top of the parallels, and the vice tightened. The parallels are then removed to avoid accidental damage – they aren’t holding the piece up, they are just an alignment aid.

When using parallels, it is important to make sure that all surfaces of the parallels, as well as the vice bottom and jaws, are free of chips or other debris. A chip underneath a parallel makes it not so parallel any more.

The vice should be tightened firmly but not too much. Overtightening can deform fragile components. It probably requires less force than you expect, though probably best to err on the side of tighter rather than having your part slip. If you can grab the part and give it a firm yank with it remaining in place, it is tight enough.

### **2.4.2 Toe Clamps**

For parts that don’t fit in the vice, they can be clamped directly to the bed with toe clamps. This is generally less convenient for a couple reasons: first, the part has to be aligned with the axes, while the vice is pre-aligned and can be used many times for one alignment; second, the toe clamps cover part of the surface, and so it may be necessary to move the clamps part-way

through. Also, they stick up above the surface and provide obstacles for the router: care must be taken that they are not run into.

Toe clamps are typically used in a bed that has T-slots in it; however, the bed on this router is just a solid sheet of PVC. A bolt goes through the slot in the clamp and then a hole in the bed. A nut (and washer!) is placed on the underside of the bed, and a step block on the stepped side of the clamp to support its back end. The tip of the clamp goes over the top of the piece, and the bolt is tightened.

The step block lets the clamp be used for various thicknesses of material. The lowest step that causes the clamp to angle downwards toward the part is the one that should be used: you want the tip of the clamp to be where the pressure is applied.

## 2.5 Materials, Feeds and Speeds

## 3 Mechanical

The frame is constructed from 45mm aluminum T-slot framing from McMaster-Carr. Additional T-nuts can be made from 3/4" wide bar stock for substantially cheaper than purchasing them from McMaster.

Each axis uses of a pair of linear guides with ball-bearing carriages and a ballscrew attached to a stepper motor. A plate of metal (or the bed in the case of the Y axis) is screwed to the bearing carriages and a piece of angle stock attached to the nut for the ballscrew. The ballscrews provide smooth motion under load, theoretically with no backlash (however these are cheap ballscrews from China, so we'll see about that). All of the ballscrews are 5mm pitch (one rotation moves 5mm).

I originally used flex-couplings to attach the ballscrews to the stepper motors, to handle the inevitable small misalignments. However, the couplings are compressible, and simply pushing on the axes was enough to move them a couple of millimeters. It is important that everything be as rigid as possible so that when the tool applies pressure, that actually goes into cutting the material rather than pushing the router around. This can cause vibration and chatter, as well as inaccuracy.

### 3.1 Tramming

There is unfortunately no provision for tramming the router – that is, adjusting the axes so that they are precisely orthogonal. There is a mechanism in the firmware for compensating for skew between the X and Y axes; however, nothing can be done in firmware if the Z axis is misaligned relative to the XY plane. There is probably some slop in the various attachments; loosening bolts, tapping with a hammer, tightening and measuring is probably the way it has to be done.

## 4 Electrical

### 4.1 Power Supplies and Distribution

There are two power supplies, one for the spindle motor and one for the steppers. The Arduino is powered via the USB connection to the host computer.

The spindle power supply takes 220V AC in (it does not work on 110V) and provides a variable 0-100 VDC to the spindle motor. The speed is controlled by a potentiometer. The spindle motor is rated at 600W. The motor came bundled with the supply.

The stepper motor supply takes either 110 or 220 VAC (there's a switch on the side); I use 220 since I already need 220 for the spindle supply. It provides 24V DC output to each of the stepper drivers.

### 4.2 Stepper Motors

There is one stepper driver for each stepper motor; this takes care of putting out the correct voltages on each of the stepper coils to make a step happen. They have four outputs, one for each stepper coil. They take 24 VDC power. There are three input signals: enable, step, and direction. The enable signal must be held at +5V for the driver to be active; this provides a convenient shutoff. The direction signal is either 0 or +5, and determines whether the motor rotates clockwise or counterclockwise. The step signal is usually kept grounded; when the motor should take a step, a pulse at 5V at least 5 microseconds long should be sent.

There are some DIP switches on the stepper drivers for controlling power output and microstepping. Without microstepping, the steppers do 200 steps per rotation. Increasing the microstepping improves resolution; increasing it too far may decrease maximum speed (step frequency would be faster than the Arduino can do the motion control for a step). If you change the microstepping, make sure to update the firmware or your dimensions and feeds will be wrong!

### 4.3 Sensors and User Interface

There are limit switches at either end of each axis. For each axis, the two switches are used in normally-closed configuration and are wired in series, so that pressing either one breaks the circuit. Since it is not necessary to know which endstop was hit (since movement direction is known), this saves three pins on the Arduino. Limit switches are used for homing and preventing out-of-bounds moves during normal operation. Hitting an endstop while milling is equivalent to hitting the E-stop button (immediately shuts off both spindle and stepper motor power).

There is a potentiometer that connects to the spindle motor power supply and varies the output voltage to control the motor speed. The voltage on the center tap is also routed to the Arduino so it can determine what the speed setting is (though it can't control it). The SSPS (set spindle speed) command will pause the router until it detects that the user has set the potentiometer to the target level.

There is an LED and a momentary-contact switch that are used for setup changes (like tool changes, work holding changes, etc.). The WUSR command will light up the LED, signalling to the user that the router is paused. Whatever changes are needed can be made, and then the button is held down for about a second to cause the router to resume.

There is also a little piezoelectric buzzer that can be made to beep with the BEEP command.



## **5 Firmware**

### **5.1 Overall Structure**

### **5.2 Communication Protocol**

### **5.3 Command Language**

### **5.4 Motion Control**

#### **5.4.1 Coordinates and Units**

#### **5.4.2 Rotation Compensation**

#### **5.4.3 Skew Compensation**

#### **5.4.4 Backlash Compensation**

## **6 Host**

## **7 Wizards**

## **8 Improvements**

### **8.1 Trimming**

### **8.2 Backlash Compensation**

### **8.3 Swarf Vacuum**

### **8.4 Work Holding**

### **8.5 Fourth Axis**

### **8.6 Tilting Head**

### **8.7 Enclosure**

### **8.8 Clickwheel**