

# Guerrilla guide to CNC machining, mold making, and resin casting

---

Home manufacturing tutorial for robot builders, model makers, and other hobbyists

Copyright (C) 2013, 2014, 2015 by Michal Zalewski ([lcamtuf@coredump.cx](mailto:lcamtuf@coredump.cx))

## 1. Preface

---

Hello there, stranger! Let me start by explaining what it's all about - and that should help you decide if this page is what you were looking for.

### 1.1. The purpose of this guide

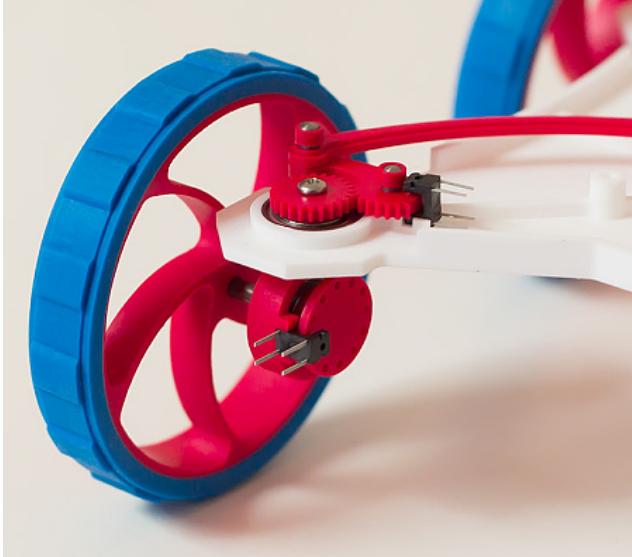
---

I'm a computer geek by day, and a hobbyist robot builder by night. I quickly learned that the craft of robotics - just like many other precision hobbies - requires either remarkably deep pockets, or a combination of outstanding manual skills and easy access to a well-equipped machine shop. Urban dwelling hobbyists who are not blessed with any of these graces tend to give up, resort to expensive and simplistic [premade kits](#), or fall back to [junkyard-quality engineering](#).

Curiously, the simplest tasks often prove to be the most challenging ones: many of us can program a microcontroller without breaking a sweat, but making a simple actuated joint, a gearwheel, or a cover in a desired shape, can be beyond the realm of imagination. For the past decade, we are being promised a revolution in desktop manufacturing - but unbeknownst to many, a simple, affordable, and home-workshop-friendly solution is already well within reach. The only problem with [computer numerical control](#) is that the workflows and materials suitable for small scale, hobby engineering are almost completely undocumented, and quite difficult to discover on your own.

Several years ago, I took a [huge leap of faith](#), decided to buy a small CNC mill ([Roland MDX-15](#)), set up a resin casting workshop, and invested months of intermittent trial, error, and triumph to understand and befriend both technologies - and document them so that others don't have to go through all the pain. It was well worth it, to be sure: I can now routinely crank out [remarkably cool](#) and precise designs in no time, and with only minimal cost:





The approach also works for others; [this hybrid legged robot](#) by Marc Hamende ([video](#)) is one of my recent favorites.

If this sounds interesting, and if you are willing to spend around \$2,000 to set up a CNC workshop, simply read on. Conversely, if you care only about resin casting or other non-CNC parts - you can skip the first chapter and go directly to one of the later bits:

- [Section 3: A crash course in CAD and CAM.](#)
- [Section 4: A primer on high-performance casting resins and moldmaking work.](#)
- [Section 5: A library of components for electromechanical projects.](#)
- [Section 6.1: Simple part design techniques for optimal rigidity and strength.](#)
- [Section 6.2: An introduction to gear geometries and transmission systems.](#)

Many of these sections will be applicable to other types of projects and other manufacturing processes - including hand-made and 3D-printed designs.

Oh, to get it out of the way: note that this page is pure, unadulterated labor of love; there are no affiliate links, paid product placements, or ads anywhere in the guide.

## 1.2. A bit on CNC machining, and how it compares to 3D printing

In basic terms, computer numerical control milling is a "subtractive" method for processing raw material (*workpiece* or *stock*) with a drill-like rotating cutter ([end mill](#)) through a set of multiaxial, computer-driven movements of the cutting head. You can think of it as a robot equipped with a Dremel tool.

Some of the most basic CNC mills resemble a [drill press on steroids](#). A growing number of them is designed specifically for home and office uses, and have [more sophisticated looks](#) - but still, in comparison with technologies such as [3D printing](#), which produces 3D shapes by additive deposition, CNC machining seems pretty savage.

Alas, for now, entry-level additive FDM technologies remain fairly useless for low cost, high quality hobbyist work; this may change in the next few years with the advent of affordable SLA printers, but that revolution is yet to come. Today, CNC has a clear upper hand on at least four fronts:

- **Precision:** when I started working on the guide, [this](#) would be a good result from a Makerbot FDM printer; and [this](#) would be a more common example. The technology has [improved since](#), but still, if you compare it to [output from a CNC mill](#), the difference becomes obvious.

CNC mills often boast mechanical resolutions of 1 μm (0.001 mm) in all axes, and produce silk smooth surfaces for almost arbitrarily complex parts. Right now, the real-world accuracy of a \$2,000

mill is usually difficult to replicate with any printer that costs less than \$40,000. In fact, even with very expensive printers, there tend to be significant constraints on what you can make: features smaller than 0.7 mm in the X-Y plane, or under 0.1-0.2 mm in the Z axis, are often taboo.

- **Choice of materials:** CNC machines work in cheap, commodity materials - anything from wax, to transparent plastics, to PCBs, to wood, to plaster, to aluminum, to steel. 3D printers, on the other hand, are restricted to a small set of proprietary stock, and produce parts that are rarely suitable for demanding applications - chiefly due to poor strength.

- **Cost per part:** With the processes advocated in this guide, the cost of making a small part out of high strength, engineering grade plastic, or from tough rubber, seldom exceeds 25 cents; the raw material hovers at about \$15 per liter or so ([this photo log](#) has a detailed price breakdown).

In comparison, the materials used by higher-definition 3D printers oscillate around \$300 to \$500 per liter; materials for Makerbot and other low cost FDM extruders are closer to \$50 - but then, their output is seldom salvageable for precision work.

- **Speed:** A typical cutting process takes between 5 minutes and 3 hours on a medium size CNC mill. When it's done, it's done: you don't need to remove intricate supports, wash, sandblast, polish, seal, or post-cure the part. Most additive technologies take much, much longer to produce usable parts of similar size, and almost always require painstaking manual work to get rid of manufacturing artifacts.

On the flip side, the fundamental advantage that 3D printers have over CNC mills is that they can often produce basic internal geometries inside enclosed parts - whereas milling can only process the areas for which there is a sufficient cutter clearance. It is not trivial to do that on low-cost 3D printers, though - and the constraint of CNC mills is seldom a big deal: almost everything can be split into halves, rotated, or so. CNC machining is the staple of industrial manufacturing. Keep in mind that almost every mass-produced item - from iPads to soda bottles - copes with this reality in a graceful way.

### 1.3. The virtues of resin casting

---

Resin casting is a nifty process that involves creating a mold, and then pouring a liquid, two-component polymer resin into the mold cavity. Once the resin sets, the finished part is removed - and the mold can be used again. This method of replicating parts is popular artists of all trades, but is seldom employed in amateur CNC or 3D printing work. So wait, why am I even bringing this up?

Well, despite sounding needlessly complicated, the overhead is actually pretty modest - and the process has some interesting advantages over directly machining (or printing) the desired shape:

- **You can use just about any material, including some truly exotic choices.** Sure, CNC mills can cope with almost everything you throw at them - but you need to obtain a suitable stock, and then tailor the process accordingly. Some materials can be difficult to find in exactly the variety and shape you need. Others can be challenging to machine quickly, especially if you're just making your baby steps with the technology: for example, machining carbon fiber composites or soft rubbers is a real pain.

The casting process gives you the option of avoiding the hassle whenever you want: master the workflow for one easily machinable, high-fidelity tooling board (or print your parts in flimsy plastic, or sculpt them in clay); and then make the final part out of rubber, glass-filled composite, low-melt metal, or reinforced concrete laced with rocks.

- **You can customize the appearance of the part, practically for free.** There is no need to spend money on a variety of differently colored materials just to make your projects look good. Add several drops of a dye to change the color of a common resin at a whim: want water clear? Opaque turquoise? Or fluorescent pink? No problem at all.

- **You can replicate and tweak parts easily.** Need to make 50 identical gearwheels? Need to replace a broken part? Need to change the impact strength or flexibility of a component? Switch to a rubber that is a bit more squishy or a bit more firm? Just mix the resin and pour it into an existing mold. The parameters of the liquid system can be easily adjusted - again, at virtually no extra cost.

- **You can conserve materials and time.** Directly machining parts involves starting with a block of material, and then removing most of it to get the desired shape. This can be wasteful. Mold cavities, on the other hand, can be quickly milled on an almost arbitrarily shaped piece of material - and when no longer needed, can be filled with putty or simply shaved away to make room for the next project. You also don't have to worry about supporting the geometry through the milling or printing process - the bottom of the mold cavity inherently does that job.

Best of all, when done correctly, resin casting is essentially pixel-perfect, providing copies with amazing dimensional accuracy and picking details so fine that you can only see them under a microscope.

If you want to do any real engineering work in plastics, it is a very good idea to [rely on resin casting](#) for a good chunk of your projects - and consequently, this guide is (in part) geared toward this goal.

## 1.4. But how much will this *really* cost me?

---

As with any hobby, sky is the limit :-) That said, if you want to get good results on a budget, the breakdown of expenses is roughly this:

- **CNC mill:** \$900 to \$3,000 for a sensible unit; around \$150 for end mills and collets.
- **CAD / CAM software:** \$0 to \$1,000, depending on taste. Free of charge options are available, and work reasonably well.
- **Resin casting setup:** \$100 for a vacuum pump; \$70 for chamber and hoses; less than \$100 for mold releases, cups, and other auxiliary supplies.
- **Media for moldmaking work:** \$230 for a giant (150 x 50 cm) prototyping board; \$120 for two gallons of polyurethane resin; \$130 for a pail of silicone. This should last for at least a year.
- **Other tools:** \$70 for a micron-resolution micrometer; \$50 for a micron-resolution dial indicator; \$40 for a simple magnifier.

For the processes advocated in this guide, the ongoing maintenance costs for a CNC mill are negligible: after several thousand hours of machining, you may have to replace several bearings or spindle motor; cutters should last for hundreds of hours, too. The savings add up quickly, especially if you are already paying for specialty or made-to-order parts.

## 2. Setting up a CNC mill

---

Okay, still interested? Let's dive in, then. The first "proper" section of this guide deals with shopping for a mill and understanding its operating characteristics; picking the appropriate cutting tools; and monitoring the performance of your setup to achieve perfect results every time.

Again, if you're using another manufacturing process and are interested strictly in the CAD tutorial or the resin casting bits, feel free to skip ahead.

### 2.1. Picking the right machine

---

General purpose, benchtop-sized CNC mills start at around \$600 and go up to \$20,000 or so. Somewhat surprisingly, quite a few of the sub-\$2,000 devices are already perfectly sufficient for most jobs, and certainly for the workflows discussed in this guide. In fact, even the most picky shoppers can get all the useful features under \$5,000; past that point, you are paying for functionality of relatively little significance to light-duty hobby work.

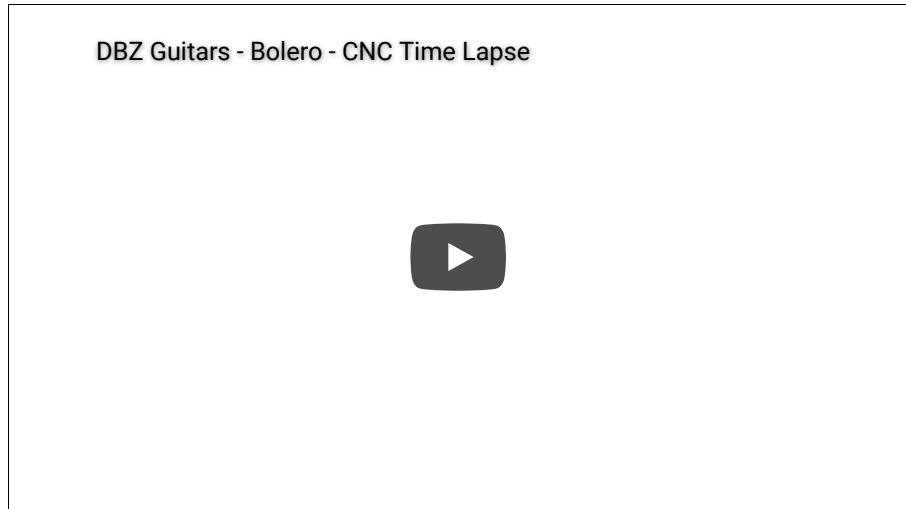
There are numerous manufacturers of benchtop CNC mills around the world; some of the best-known brands include [Roland DG](#), [Sherline](#), [Taig](#), and [Syl](#). But be sure to search around; there are quite a few other companies that cater to local markets - say, [Probotix](#), [Deepgroove1](#), [LittleMachineShop](#), [Romaxx](#), [Microkinetics](#), [MAXNC](#), [Microproto](#), [Light Machines](#), [Minitech](#), [Flashcut](#), [Tormach](#), [Smithy](#), [ShopBot](#), [Torchmate](#), [CNC Masters](#), [ACT](#), [Charlyrobot](#), [EasyCut](#), or [Laguna Tools](#). Some people had luck with [ultra-low cost mills made in China](#), too.

Of course, it is also possible to build your own machine from scratch. Doing so is not necessarily economically sound, because there is a significant price tag attached to high-precision linear motion systems, machine spindles, and powerful servos or stepper motors; on top of that, you will probably have to iterate through several designs, and the project will consume several months of your time. Still, if you are so inclined, there are low-cost plans and kits available on the Internet.

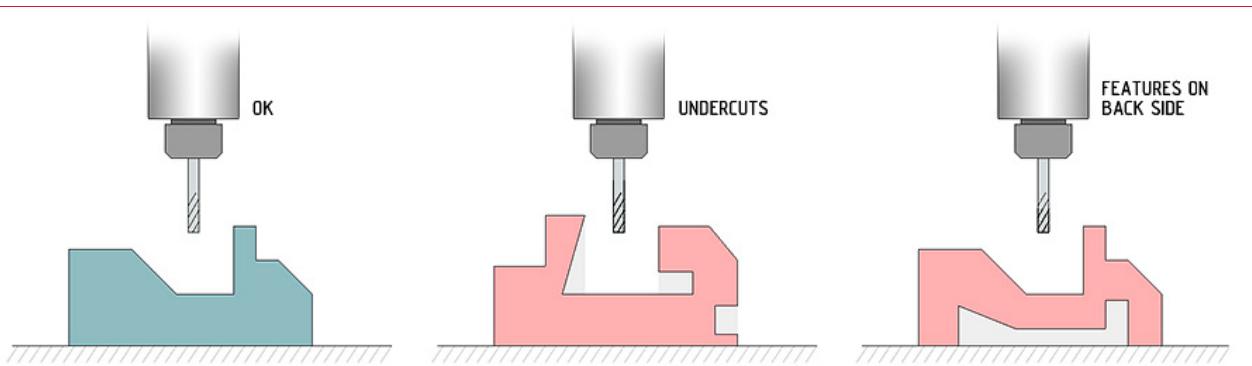
When shopping for a pre-made system, there are several key characteristics to pay attention to; let's have a look at them, and use them as an excuse to discuss some of the inner workings of CNC milling jobs.

### 2.1.1. Number of axes

This is perhaps the most fundamental quality of any CNC mill. In the most basic design, the cutting head can move in three directions - X, Y, and Z - and the tool itself always points down, aligned with the Z axis. In this setup, the machine can only machine shapes that can be represented using a two-dimensional "depth map" projected onto the workpiece: the cutter may descend lower for some X-Y coordinates, and move up for others, but it will not enter the workpiece from any other side. This video is a pretty good illustration of the process:



In this machining mode, the machinable geometries are outlined here:



**Note:** CAM applications are designed to fail safely; that is, if any of the features of the model cannot be reached without plowing through another essential section of the geometry, the problematic region simply won't be machined at all. The gray regions in the two workpieces on the right correspond to the material that will be left in place.

The limitations of three-axis machining may seem severe, but seldom truly are. Every section of an industrial injection mold or a metal forming die typically needs to be a depth projection anyway, so that the processed material could be pulled out of it easily. Even in direct machining, it is common to simply flip the workpiece with the aid of registration pins. [This video](#) illustrates the manual rotation process fairly well.

That said, there are some shapes that truly benefit from automated, multi-directional machining; this includes exotic types of gears (helical, herringbone, and worm geometries) and certain categories of jewelry (say, rings). For these uses, some CNC mills come with additional rotary "axes": the so-called A axis corresponds to rotating the workpiece around the X axis ([see video](#)); B axis stands for rotation around Y; and C axis is the rotation around Z. The four-axis AXYZ setup is the most common one.

The premium for fourth axis starts at around \$100 for manual indexers (a precision rotary chuck that holds the workpiece, but where the angle needs to be dialed in manually); and from \$500-\$1,000 for computer controlled units.

**What to buy:** 99% of your moldmaking work will not appreciably benefit from a fourth axis, so three axes are perfectly fine. You may want to get a mill where fourth axis is an option, though, especially if you are also planning to do artistic work.

### 2.1.2. Mechanical movement ranges

---

Greater X-Y-Z tool movement ranges translate to the ability to make larger parts in a single pass. It's important to pick a mill that won't get in the way of your imagination - but to make this call, you need to calibrate your expectations sensibly.

As an extreme example, let's consider building a man-sized biped robot. You don't need a man-sized mill for that job - for at least three reasons:

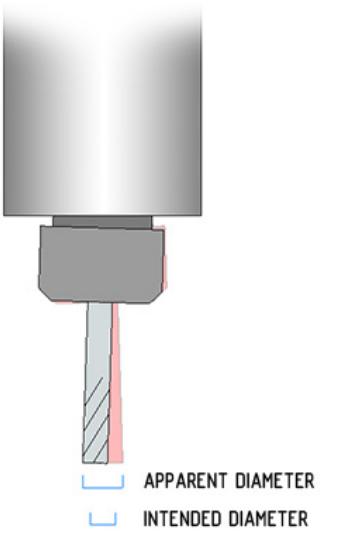
- Although the entire project would be obviously pretty big, virtually all the individual parts should be much smaller. In fact, most of the components will likely fall into the range of 1 to 30 cm.
- Any oversized part can be broken into sequentially machined segments. If you are using the resin casting process, this scales up almost arbitrarily; in other types of work, you will see some (usually modest) constraints.
- Many large and simple elements do not have to be machined at all; for example, it would be sensible to make the frame out of pipes or metal rods sawed to the desired length, and then simply machine custom connectors and joints that hold the frame together.

**What to buy:** do your own math. In my experience, about 15 x 10 cm in the X-Y plane is a good starting point, and about 30 x 20 cm will accommodate almost any medium-size robotic job. In the Z axis, you will probably not need more than 4 cm or so; and going over 8 cm is usually pointless. Whatever you do, do not confuse movement ranges with table dimensions, though.

### 2.1.3. Use of a specialized spindle

---

Spindle - the part that connects the motor and the rotating tool - has a profound impact on the accuracy of any CNC mill. Its role is to ensure that the rotation of the tool is highly concentric and vibration-free, and that it stays this way under load. If the whole rig is not perfectly centered, you may end up with a situation such as this:



The total amount of back-and-forth wobble - in other words, the difference between the intended and effective diameter of the tool - is known as *total indicated runout*, or TIR. High TIR will not only affect the dimensional accuracy of machined parts, but will also ruin surface finish, and prematurely wear the tool. In fact, the effect is pretty dramatic: in some materials, eccentricity of 0.01 mm can reduce tool life by 50%.

Proper CNC spindles are usually long, round or rectangular blocks of metal with precision ball bearings mounted on both ends (and often pre-tensioned with a spring). Inside, there is a heavy-duty rotating shaft that couples the motor belt drive system to the tool holder. With quality spindles, TIR usually can be kept below 2  $\mu\text{m}$ .

Some of the low-end manufacturers don't bother with a proper spindle, however; the most common example of this are CNC mills that use repurposed [manual rotary tools](#). These cases are a bit of a gamble: some of them may have still somewhat bearable TIR in the vicinity of 0.01 mm - but some will be as bad as 0.10 mm, which makes them completely useless for precision work. Runout aside, you also can't be sure if the tool is perfectly aligned with the Z axis or not; if it isn't, that opens a yet another can of worms.

**Note:** to put all these numbers in perspective, 0.10 mm is roughly the diameter of a human hair; level differences of this magnitude can be easily felt when sliding your finger across a hard surface. Notches down to about 0.05 mm can be easily seen on smooth but matte finishes - and on glossy surfaces, the threshold may be closer to 0.01 mm or so.

**What to buy:** try to avoid CNC mills without real spindles; if you need to get one, ask the manufacturer about TIR. If they are not sure, it's an obvious red flag: the parameter can be trivially measured with a [\\$50 tool](#), and is one of the most rudimentary things to examine when designing a mill. Note that there are [aftermarket spindles](#) that can be fitted into certain mills, though!

#### 2.1.4. Movement precision

There are many factors that contribute to the real-world precision of a CNC mill, but one of the most important aspects is repeat accuracy: the ability to return to the same position over and over again. Along with spindle characteristics, this quality has a tremendous impact on surface finish, and on the dimensional accuracy of small parts.

Repeat accuracy is affected chiefly by two things:

- **Backlash:** many types of transmissions will have some amount of play, which often leads to imperfections in positioning. This play tends to be greater in CNC mills that use stepper motors or [acme screw drives](#); and lower with servo motors and [ball screws](#).
- **Mill rigidity:** machines that use low-cost materials (HDPE, plywood, sheet metal) to hold everything together, rely on rudimentary slides to support the moving parts, or use timing belts as a motion

system, will deflect more significantly under load, or in response to own acceleration and deceleration. This problem is much less pronounced in mills with heavy-duty cast frames and ball-bearing linear ways.

Unfortunately, there is no widely accepted standard for testing repeat accuracy; many manufacturers don't bother to advertise it, and others test it with varying levels of honesty. In fact, the good guys will give you a figure that represents the worst-case, momentary deviation following a rapid long-distance movement - but that's not really representative of most types of fine work with sub-millimeter tools.

Now, don't despair: the good news is that most of the commercially available mills are actually pretty good in this department, especially when moving slower and doing precision cuts in easily machinable stock. You can expect many entry-level mills to conduct themselves within 0.02 mm or better during normal work; more expensive units with ball screws and servo motors will probably stay around 2-5  $\mu\text{m}$ .

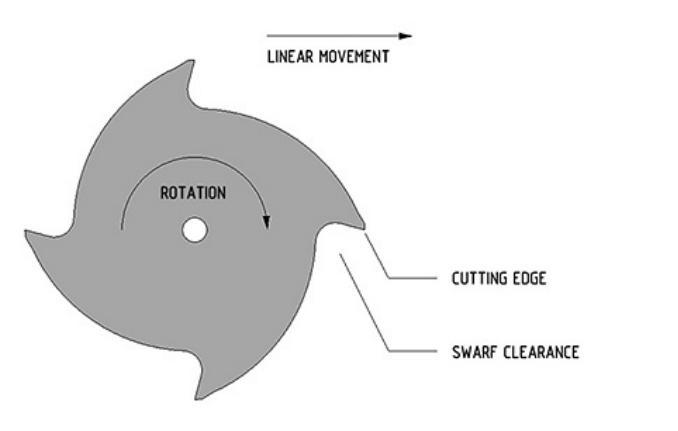
Accuracy aside, mechanical resolution is the other important piece of the puzzle. Stepper or servo motors in a CNC mill can assume only a certain number of positions per turn, and that translates to a specific minimum distance by which the table or the cutting head can be moved around. Insufficient mechanical resolution means that the mill will have difficulty smoothly approximating certain curves, and may end up producing unattractive finish.

**What to buy:** The basic rule is that you should not expect a plywood-based contraption with acme screws to reach 1  $\mu\text{m}$  repeat accuracy. If the manufacturer advertises an improbable value, ask them to explain. If they advertise a suspiciously high figure (over 0.1 mm or so), be wary, too. As for the mechanical resolution: look for 5  $\mu\text{m}$  or better.

### 2.1.5. Machining speeds

Time is money. When it comes to CNC machining, the time needed to complete a job is to a surprising extent dependent on your skill and the capabilities of your software - but with a skilled operator and good toolpath decisions, the final part of the equation is always the performance of the mill itself.

To understand how the mill's performance is tied to the numbers you see in the datasheet, it is helpful to look at the geometry of a typical end mill. Upon closer inspection, the tool closely resembles a drill: it consists of a round shaft with several blades (flutes) wrapped around it in a spiral fashion. As opposed to a drill, however, these flutes have a sharp, exposed edge running along their entire length; this is because the bulk of their work is meant to be done by moving sideways. This is how it looks from the top:



Even in the most easily machinable material imaginable, the cutter is able to scoop away only a certain amount of swarf per turn - just enough to fit under the flute. If you exceed that capacity, you will end up dragging a clogged, non-cutting tool across the workpiece - which ends with one or the other eventually giving up.

For every material and cutter geometry, there is an optimal ratio of linear speed and cutter RPM that leads to efficient, high-quality machining. This is often expressed as *feed per tooth*. In plastics, the ideal values are:

- Up to **0.1-0.2 mm** for larger tools used to do the bulk of material removal in every cutting job. These tools frequently have four flutes, so this translates to 0.4-0.8 mm per turn.
- Down to **0.01 mm** or less for sub-miniature, 2-flute cutters used to reproduce minutiae detail (0.02

mm per turn).

In practical terms, it's healthy to aim for mills where the ratio between maximum movement speed (mm/min) and maximum RPM hovers around 0.4 to 0.8 for optimum performance during rough cutting. At the same time, there is also some value in shopping for the highest maximum RPM you can get - as it lets you move faster during the precision finishing steps.

Of course, there are some gotchas:

- Very high spindle speeds tend to be problematic; going over 25,000 RPM or so may cause problems with heat dissipation, tool oscillation, and so on.
- High traverse speed can help even without high RPM. This is because certain types of cutting jobs involve a fair amount of rapid, non-cutting movements between various locations to be machined. Good CAM applications should know to avoid excess travel, but not all CAM applications are very good.
- Acceleration matters. Fast-moving mills need to be good at accelerating and braking to fully utilize their capabilities during series of short, stop-and-go movements that are often involved in CNC work. Rates of at least 0.1 G (~1 m/s<sup>2</sup>) are desirable in any mill capable of four-digit speeds.

*What to buy:* at least 6,000 RPM is nice; and if the aforementioned speed ratio is favorable, there are no real downsides to going up to 20,000 RPM. Maximum movement speed, in mm/min, should be ideally at least 6-10 times the movement range, so that it doesn't take more than several seconds to traverse the table.

## 2.1.6. Tool sensor support

Spare for some pathological situations, the mill is intrinsically aware of the position of its spindle at any given time; but the actual cutting action takes place beneath the spindle - at a distance dictated by how far the tool sticks out from its holder.

And here lies the problem: most toolholding systems do not allow you to precisely preset tool extension length, or to maintain it when you replace the cutter. If you switch the tool in the middle of a machining process, and don't compensate for the difference, the results will be off; in fact, the tool may unexpectedly hit an uncut area and break.

There are several manual tricks that can be used to work around this issue. One of them is to place a thin strip of paper or foil in a fixed reference location, and then slowly lower the tool until the strip gets caught between the cutter and whatever happens to be underneath. By comparing the Z position of the spindle at that point with the reading obtained for the previous tool, the appropriate offset can be calculated and communicated to the machine. But of course, this technique is somewhat inconvenient, and accurate only to perhaps 0.05 mm.

A better approach is to incorporate a tool height sensor into the mill. The sensor is typically just a flat block of precisely machined soft metal; the mill automatically lowers the tool onto the sensor until contact is made - which, in the simplest design, is detected by noticing the flow of current between the probe and spindle body. The accuracy of this approach is often better than 0.01 mm.

*What to buy:* try to find a mill that has a built-in sensor, or can be equipped with one. Failing this, you can always rig a manual tester that uses the same operating principle, and simply illuminates a LED.

## 2.1.7. Tool mounting method

The spindle must be terminated with some sort of a tool holding device. There are several common systems that you will bump into when shopping around:

- **Jaw chuck:** this is the design familiar from drills and other hand tools. These devices tend to offer poor concentricity (TIR can be as high as 0.1 mm) and should be avoided if possible.
- **Fixed diameter chuck:** essentially a metal sleeve with its inner diameter matching the size of the tool. It is operated with a set screw or with heat (shrink-fit). It's pretty accurate, but can make tool changes costly or cumbersome. For example, in Roland MDX-15, you have to replace the entire spindle assembly to switch between two different diameters of the shank.

- **R8 collets:** these are popular in the States, cheap, easy to use - but because of the way they grip the tool, they usually have relatively poor TIR (0.01 to 0.02 mm). That's a bit high for comfort.
- **Morse taper (MT3) collets:** a superior alternative to R8, typically with TIR below 5 µm.
- **ER collets:** probably the best option for small CNC mills. These collets can accommodate a wider range of tool shanks, and the "ultra-precision" variety from reputable manufacturers tend to have TIR under 2 µm.

*What to buy:* If you can get ER16 or MT3, go for it. Otherwise, just make sure that the toolholding system is versatile enough to accommodate common shank sizes (3, 4, and 6 mm for metric cutters; 1/8" and 1/4" for imperial system tools), and will be sufficiently precise for your needs.

### 2.1.8. Availability of CAM software

---

Manually programming your CNC machine is about as much fun as building a steamboat out of toothpicks. For a higher-level approach, you need to turn to [CAM software](#): it automatically analyzes the provided geometry (created with any 3D modelling application) and converts it to a set of paths that need to be retraced by the tool to approximate the desired shape. Once these toolpaths are ready, the software then breaks them down into a sequence of painfully basic instructions that actually make sense to the controller embedded in the mill; say, "set speed to 12,000 RPM" or "move cutter to X = 10.245, Y = 5.000, Z = -2.000".

The toolpath generation stage is largely hardware-agnostic; but the program generation one isn't. It's good to shop for a machine that speaks a common and well-documented language - or, lacking this, is popular enough to be supported by some of the best-known CAM apps. Keep in mind that even if the manufacturer bundles the mill with some starter software, you don't want to be left out in the cold if the application one day refuses to work with your new PC - or if it simply turns out to be of poor quality.

The most common quasi-standard language used by almost all CNC mills is called [G-code](#) (aka "NC"). Calling it a real standard may be a stretch: there are very significant variations in how the syntax is implemented by the manufacturers. Still, having support for G-code spells rudimentary compatibility, or at least easy integration, with almost any CAM application on the market. For other languages, this is not always given.

*What to buy:* check if the mill is supported by common third-party packages ([Deskproto](#), [VisualMILL](#), [madCAM](#), [MeshCAM](#), [Mayka](#), etc); if it's not, and if it speaks something else than a clearly documented variant of G-code, be wary.

### 2.1.9. Size, weight, power needs

---

We're almost done: the last thing to do is a quick reality check. Benchtop mills span from units no larger than an inkjet printer, to ones weighing in excess of 200 kg and taking up almost 1 x 1 m of desk space. When shopping for the larger models, be sure to account for their physical characteristics, and make sure you have a way to get them in your workshop to begin with (some doors are barely 70 cm wide).

For heavier mills, it is also important to have a piece of sturdy furniture; it's not just the static load that you have to worry about, as the machine may also produce horizontal shear forces due to acceleration and deceleration of the cutting head. Not every wobbly desk from Ikea can handle that - but any proper workbench should.

Benchtop mills usually run on standard, single-phase 110 / 230 VAC power supply, but of course, make sure to double-check. They may require several amps in peak, so you don't want them to share a single circuit with a vacuum cleaner, an electric kettle, or a space heater - especially in an older home.

### 2.1.10. And now, all the things you don't have to worry about

---

Okay - that sums up the list of parameters that are worth looking at. There are also some characteristics that sound important, but usually aren't - so to help you decide, here's a quick list to consult when in doubt:

- **Positioning accuracy:** this value tells you how true the machine's idea of a millimeter is to a calibrated reference. The difference is usually negligible, and even if it wasn't, you can compensate in software without any special effort.

- **Origin reproducibility:** this indicates how good the machine is about resuming the same starting location after being power cycled. Not particularly important in normal work, unless your power goes out in the middle of a cutting job.
- **Software resolution:** unlikely to be lower than mechanical resolution, and fairly meaningless if it's higher.
- **Motor power:** spindle motors delivering as little as 75-100 W should be perfectly sufficient for the processes advocated in this guide. More powerful mills are beneficial chiefly in heavy-duty metal work.
- **Table size:** easy to confuse with the actual machinable area, but nowhere near as important. The difference between a 2 cm margin and a 10 cm one usually doesn't matter at all.
- **List of millable materials:** every mill should be able to cope with a wide variety of materials; don't read too much into the list provided by the manufacturer. The only real difference is that some units are not rigid or powerful enough to handle steel. (Some people insist that smaller and less rigid machines that can't machine steel should be called "CNC routers" and not "CNC mills", but this is not a particularly well-established practice, and other definitions exist.)
- **Tool changer (ATC):** it's a convenient device for unattended factory jobs, but an expensive and unnecessary gimmick in hobby work.
- **Contact scanner accessory:** contact scanning heads are a gadget available for some CNC mills, but chances are, you will use them to scan a couple of coins or pendants, and then lose all interest. For reverse engineering of mechanical designs, just using a caliper works a lot better, and takes less time.
- **Noise ratings:** probably not what you think. These parameters describe the noise produced by the mill when in standby mode (spindle off); or when operating, but not making contact with a workpiece. Benchtop mills are fairly quiet in no-load conditions; and even the most compact and underpowered devices can be pretty loud when actually machining something.

The actual noise level will range from barely perceptible when working in waxes or using sub-mm cutters, to pretty unpleasant when rapidly plowing through metal with a large-diameter tool. In fact, just imagine the noise made by a saw or a drill.

Well, that's probably it. If you spot any other puzzling parameters, please let me know.

### 2.1.11. So, which one should I buy?!

---

That really depends on your budget and the scale of the projects you want to be working on. Here is a list of some of the fairly popular, inexpensive mills, along with their catalog prices. Many of them sell for around 15% less if you talk to the right distributor:

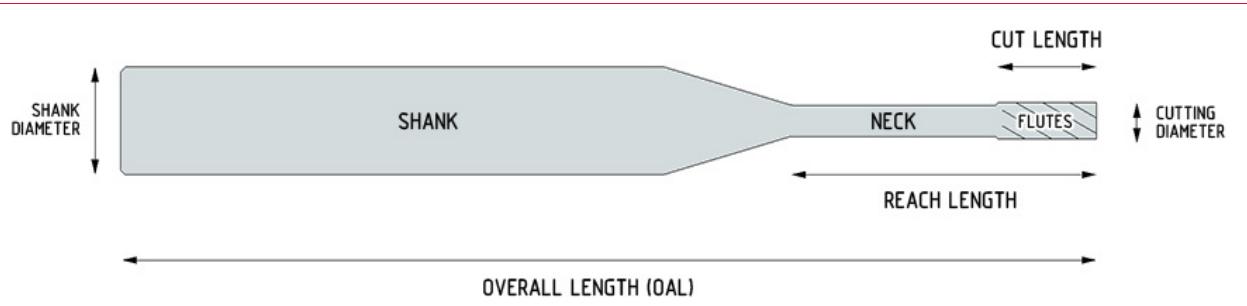
- [Chinese "3020" mill](#) - around \$500.
- [Roland iModela](#) - \$900 (no longer available in the US?).
- [Probotix FireBall V90 + several options](#) - \$1,300.
- [MAXNC-10](#) - \$1,600.
- [Deepgroove1 CNC mill](#) - \$1,700.
- [Sherline 5400 + CNC package](#) - \$1,800.
- [Taig 2026ER](#) - \$2,300.
- [Nomad 883](#) - \$2,500 (preorder, looks very promising).
- [MicroMill DSLS 3000](#) - \$3,000.

- [Flashcut 2000](#) - \$3,000.
- [Roland MDX-15](#) - \$3,400 (discontinued, replaced by SRM-20).
- [LittleMachineShop 3501](#) - \$4,900.
- [Roland SRM-20](#) - \$4,900.
- [Syl X4](#) - \$5,800.

A good place to search for user opinions is [CNCzone.com](#). Good luck!

## 2.2. Stocking up on end mills

Ordering a CNC machine? Well, the next stop is getting some cutters. The selection available on the market is quite overwhelming, so to save you time and money, let's talk about some of the properties that set these tools apart. Oh, before we dive in... here's a drawing of a typical end mill, and all the lingo you will have to memorize soon:



All right - so here are the differences you will see:

- **Material:** end mills are made either out of cobalt steel alloys (known as *high speed steel*, or HSS), or from tungsten carbide in a cobalt lattice (colloquially shortened to "carbide"). The latter option is considerably harder, more rigid, and more wear-resistant - and for the tool sizes we are interested in, carries a relatively small premium (30% or so).

**What to buy:** Stick to carbide.

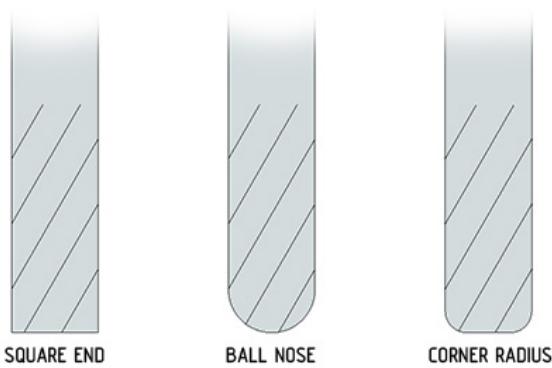
- **Coatings:** carbide cutters may be further coated with ceramics such as [titanium aluminum nitride](#) (TiAlN, aka AlTiN), titanium nitride (TiN), titanium carbon nitride (TiCN), or with amorphous or crystalline diamond. The coatings tend to improve hardness or reduce friction; the bluish-gray TiAlN coating is probably the most common one. In non-abrasive plastics, this particular coating doesn't have a very pronounced effect, but it extends tool life by some 20% or so - and the cost is only 10% extra, so it's not a bad deal.

Of course, in some situations, tool wear is a more pronounced concern; for example, glass or carbon fiber composites can leave a mark on carbide tools in a matter of hours. In the same vein, in ultra-precise micromachining, you may have to worry even about early-stage wear, because it can subtly affect the dimensions of the tool. In such cases, amorphous "diamond" coatings offer more definite benefits compared to TiAlN, often extending tool life by a factor of 2-3x. Alas, the coating at a 30%-40% premium and is available only for some tools. (True, crystalline diamond works, too, but costs a lot more and is reserved for more demanding jobs.)

**What to buy:** If you're on a tight budget, skip any coatings and pocket the change. Otherwise, TiAlN or amorphous diamond is probably not a complete waste of money, although you can still skip it on your first set of "training wheel" tools. Stay clear of more pricey coatings: crystalline diamond will probably never pay for itself in our work.

- **Tip geometry:** precision machining operations in plastics will almost always rely on "vanilla", single-end finishing tools with no chip-breaking ridges, no taper, no coolant outlets, and so forth. The tips of

these standard tools come in three basic flavors:



Flat tip cutters are the primary tool in all sorts of mechanical work. Ball nose cutters come handy when you need to reproduce gentle slopes and other organic designs. And lastly, corner radius cutters offer a compromise design that combines 90° arcs in the corners, and a flat mid-section - but you can safely ignore them until you have a specific itch to scratch. About the only other type of a cutter worth mentioning here are conical engraving cutters, which look [like this](#); these are useful for making very fine but shallow cuts, for example when machining traces on a PCB.

**What to buy:** build a competent collection of flat tip cutters first. If you want to experiment, grab one or two ball nose cutters, but you probably won't be using them that much in non-artistic work.

- **Cutting diameter:** end mills come in cutting diameters from 0.01 mm (a lot thinner than [human hair](#)) to 50 mm and more. For most intents and purposes, cutters below 0.25 mm or so are just not very useful, unless you are doing some truly hardcore micromachining; and over 8 mm or so, they get too big to mount and meaningfully use on benchtop mills.

Your cutters need to strike a sensible balance: large tools can't machine tiny features, but small tools remove very little material in every pass, making them useful mostly for selective refinishing work. In fact, a two-fold reduction in tool cutting diameter often translates to a 10-fold reduction in effective material removal speed!

**What to buy:** try to get a 3 mm (or 1/8") cutter to handle most of the machining work; for large projects, 6 mm (or 1/4") may come handy, too. On top of that, keep 0.4 mm and 1 mm cutters for touching up fine details.

- **Cutting diameter tolerance (tricky!):** "1 mm" cutters don't always have a diameter of 1 mm. In a nod to the legacy of drilling and tapping applications, many manufacturers make their tools to non-symmetric tolerances, as denoted either with a pair of explicit values (say, +0.00 / -0.04 mm), or with an [ISO tolerance code](#) such as "e8".

Of course, modern tool grinding machines are often a lot more precise than that; micron accuracy is not unheard of. The machine is therefore preset to simply crank out tools at the mid-point of the specified range. Using the example of a 1 mm tool specified as +0.00 / -0.04 mm, the actual diameter will end up being 0.98 mm or so. In most uses, you won't even notice - but then, every now and then, such a subtle difference can bite back.

Okay, so the case of explicit tolerances is pretty clear; as for the "e8" case: for cutters with diameters below or equal to 3 mm, the offset is -0.021 mm. For cutters over 3 mm and up to 6 mm, the difference is -0.029 mm.

**What to buy:** whatever you want, just make sure to double-check the tolerance, and write down the actual diameter of the tool. Also, if the vendor declares a suspiciously sloppy tolerance, approach such tools with caution, and perhaps ask the company if they mean it.

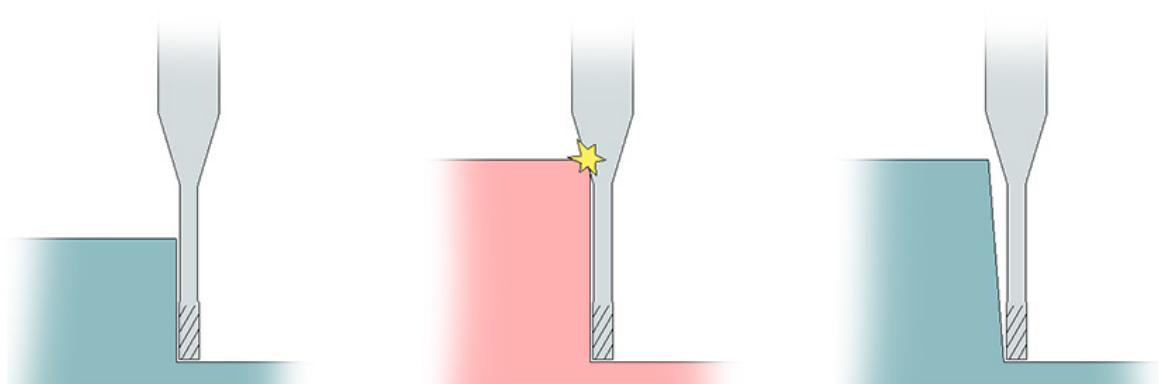
- **Shank diameter:** this is the diameter of the top portion of the end mill - the one that goes into the

holder. It has no significant effect on the cutting process, so you simply need to make sure that you have a matching collet for all the tools you intend to buy.

- **Cutting length:** the cutter will have flutes extending only through some of its length. The height of the cutting region sets the upper limit of how deep the tool can be driven into the workpiece in a single pass.

*What to buy:* for the processes outlined here, you typically only need the flutes to extend to a distance equal to about 50% of tool diameter. Because the flute-bearing section somewhat more vulnerable than the rest of the end mill, and may also be subject to uneven wear, it's actually beneficial to opt for stub-flute tools when you have a choice.

- **Reach length:** many end mills have a neck - a section between the flutes and the shank, across which the diameter of the tool does not exceed that of the cutting tip (and in most cases, is about 10% percent smaller). The distance from the tip of the tool to the end of this section puts an absolute limit on the depth of machining near any vertical walls:



As you can see in the rightmost drawing, the problem can be avoided by introducing a non-zero draft angle: a subtle slant on any tall, vertical walls. Draft angles also help with removing cured parts from rigid molds - but are not always desirable or practical. That's why having a tool with a decent reach length is still a good plan.

On the flip side, more is not always better: with miniature long-reach tools, as the neck gets longer, the cutter becomes more prone to deflection and easier to break. A 1 mm tool with a 3 mm reach can be operated with few worries; at 10 mm, you need to be a bit more careful; and at 30 mm, it starts to require ninja skills.

Oh, of course, some tools don't have a neck to speak of; in these cases, there is a taper that immediately follows the flutes, and reach length is identical to cutting length. That may be good or bad, depending on whether flute length is sufficient for your needs.

**Note:** there is an interesting special case - some 3 mm or 6 mm tools have shank diameter identical to their nominal cutting diameter; the same goes for their imperial 1/8" and 1/4" counterparts. You can think of them as having reach length equal to total tool length, minus about 20 mm needed for the collet.

There are some drawbacks of driving these tools deeper than their cutting length, though: their pretend "neck" doesn't have a reduced diameter, so when dealing with zero draft angles, it may end up rubbing against the workpiece. This usually has no effect in prototyping plastics, but quickly becomes a problem in metals and other tough stuff.

The situation gets slightly worse if the actual cutting diameter is smaller than the nominal one, due to non-symmetric tolerances. In these cases, driving such a tool too deep may have a minor but noticeable impact on dimensional accuracy or surface finish.

**What to buy:** moldmaking often involves relatively deep cuts and vertical walls. Make sure that your 3 mm tools have a reach of at least 20-30 mm; for 1 mm, stick closer to 10 mm; and for 0.4 mm, 2-4 mm should do the trick.

- **Overall length (OAL):** self-explanatory. Typical mold cutters have lengths of about 50 to 75 mm; some lower cost or subminiature tools may be available as 38 mm, too. Longer end mills - up to 150 mm or so - are also available, but seldom worth the price.

The main significance of this parameter is that the tool will stick out of the spindle by no more than OAL, less something around 15 mm for the part that needs to be gripped by the collet. Life gets more complicated when the collet nut and the rest of the spindle gets in the vicinity of the workpiece, so it's preferable to have tools long enough to deal with the typical depth of the molds you are planning to work on.

**What to buy:** if your projects will be anything like mine, your molds will probably range from about 6 to 22 mm in depth, with infrequent excursions down to 40 mm or so. If so, OAL of 50-75 mm is a good pick.

- **Number of flutes:** higher flute count translates to faster material removal rates and superior finish, because there are more cutting passes per every turn of the spindle. Unfortunately, there are modest limits to how many sufficiently robust flutes you can cram onto a small cutter, and still have enough room for swarf; 4 flutes are the common limit, although 6 are not unheard of.

**What to buy:** for the materials we will be using for moldmaking, the more flutes, the better; at least three or four are strongly desirable on 3 mm cutters, and nice to have on smaller tools. For aluminum, rubbers, and malleable plastics, one or two flutes may be a better pick.

- **Helix angle:** zero degree flutes are perfectly straight, and run vertically along the tool. The industry standard is 30°, which offers a sensible balance between edge sharpness, vibration, and swarf removal speed. Lower angles are used to rapidly engage and remove softer materials (e.g., rubbers, thermoplastics); while higher angles improve tool life and reduce vibration in hard materials.

**What to buy:** there is virtually no difference between 15, 30, 45, and 60° when working with easily machinable plastics; you can safely disregard this parameter, and just focus on finding the most interesting and the most affordable tool.

- **Center cutting ability:** almost all the cutters of interest to CNC work have a bottom flute that extends through the entire diameter of the tool - which allows it to be plunged straight down into the workpiece, and effectively act as a somewhat competent drill. But a small minority of end mills are not center-cutting, and can't be operated this way. You probably don't want that.

**What to buy:** avoid non-center-cutting tools.

Phew! My favorite tool manufacturer is [Hanita](#) (now a division of Kennametal, confusingly sold under the brand name of WIDIA): they have an unmatched selection of metric tools at reasonable prices, and are available all over the world. If you are in the States, [Sierra Tool](#) is a good reseller; [Centerline Industrial](#) is a bit cheaper, but for some reason, refuses to ship to residential addresses. And if you are anywhere in Europe, I can strongly recommend ordering Hanita products with [ITC](#).

Hanita aside, [Harvey Tool](#) has a very interesting selection of imperial system miniature tools in the US - and I found [K&H Sales](#) to be a dependable distributor. Other US manufacturers include [OSG Tap & Die](#), [Monster Tool](#), [Micro100](#), and [Microcut](#), but their catalogs are not as impressive. Readers in the EU may want to check out [Nachreiner](#).

Here are the catalogs of the three most interesting manufacturers:

- [Hanita: full end mill range](#) (most are "e8").
- [Harvey Tool: all tools](#) (some have non-symmetrical tolerances).
- [Nachreiner: carbide finishers](#).

As for practical recommendations, I would suggest starting with Hanita 401403000, 402403000, or Harvey 73118-C4, as the baseline "3 mm" cutter (\$15-\$30); Hanita 7N2201021 or Harvey 76440-C3 for 1 mm work (\$30-35); and Hanita 7N2200410 or Harvey 992515-C3 as a long-reach ~0.4 mm tool (\$35). If you are not on a very tight budget, it makes sense to order two of each - it's easy to mess something up in the heat of initial experimentation.

With the tools selected, you also need to make sure you have the right set of collets. For ER16, if you are not desperate to save a few bucks, try [Rego-Fix](#) "UP" (ultra-precision) collets; they retail for about \$45 a pop, and are carried by [K&H](#); another good option are "DNA" collets from [Techniks](#) (about \$30, require a custom nut). You can find lower-cost collets from many other, more obscure brands - but they are not always particularly good.

Oh, one more thing: for ER16, every collet has a specified clamping range - for example, 3.00-2.00 mm. It is always preferable to use the upper value: a 3.00-2.00 mm collet is better than a 4.00-3.00 one when holding a 3 mm tool.

## 2.3. Periodic testing and troubleshooting

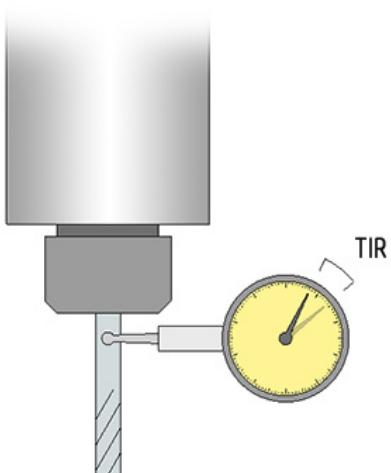
Before embracing any complex or high-precision projects, it is important to understand the performance of your mill, and see if anything needs to be fixed, adjusted, or compensated for. While CNC mills don't require constant tuning, making several simple measurements after unpacking the device can save you a lot of time. If you neglect this step, you will find that troubleshooting mill accuracy issues in complex, real-world projects tends to be a daunting task, simply due to the sheer number of variables to look at.

The essential tool that you will need to perform the initial measurements is a micron-resolution dial indicator with a magnetic base. You can get a no-name unit on eBay for about \$50 ([link](#)), or go with Mitutoyo or other reputable brand for about \$220 ([indicator](#), [base](#)). For lower-cost mills with a rotary tool acting as a spindle, you may be better served by a 0.01 mm indicator, though; in this case, you can get a Mitutoyo one for about \$80 ([link](#), base not included).

### 2.3.1. Spindle TIR

The first thing to check is the runout of the spindle and the tool holder. Wipe clean the internal spindle taper and the collet (use WD-40 if there is any excess grease or other accumulated dirt; a small [brass brush](#) works well for any stubborn gunk), and install a tool that extends at least about 20-30 mm from the collet. Tighten until you feel definite resistance, but don't overdo it - excess force may deform the collet, and is not essential in lightweight work.

Next, affix the dial indicator to the table or other sturdy surface, make the tip of the indicator touch the tool near the spindle, and observe the change in readings as you gently turn the spindle by hand, preferably at the top (the mill should be turned off, of course). Be very careful not to exert any unnecessary pressure on any of the parts.



Let's call the result of this measurement  $R_{\text{collet}}$ . Move the indicator about 10 mm lower (stay clear of the flutes) and repeat the test; we'll refer to it as  $R_{\text{middle}}$ . Finally, if possible, remove the tool and the collet, and reposition the

indicator to make contact with the internal taper of the spindle (the measuring tip now pointing up). Repeat the procedure, and write down the result -  $R_{\text{taper}}$ . Here's how to interpret the data:

- $R_{\text{taper}}$ : this measurement tells you about the concentricity of the spindle itself. If it is excessive, it may be time to service the spindle (e.g., adjust the internal tension spring; disassemble it and install a new pair of precision bearings; or, insert a shim somewhere to fine-tune the concentricity).
- $R_{\text{collet}}$ : if the spindle measurement looks OK (and only then!), this value helps you estimate the eccentricity contributed by the tool holder (on top of  $R_{\text{taper}}$ ). If the value is high, examine the collet for dirt and damage (again, WD-40 is a pretty good cleaning liquid). Small tool holder misalignment can be often "fixed" simply by adjusting the tightening torque, or applying a gentle pressure to the appropriate side of the tool when tightening the nut. If that doesn't help, replacing the collet - perhaps with a higher-quality model - may be the way to go.
- $R_{\text{middle}}$ : this value lets you estimate the eccentricity at the tip of the tool, which ultimately, is the most important observation to make. You can see if the TIR is increasing, decreasing, or staying about the same in relation to  $R_{\text{collet}}$  - and use that to extrapolate the apparent diameter of the tip.

For a quality machine with a dedicated spindle and ultra-precision ER16 collects, after some adjustments, TIR should preferably stay within 2  $\mu\text{m}$  or so; and for any mill, it is good to have runout within 0.01 mm. If you are seeing something much worse, poke around and see if you can improve it: it's usually simpler than it may seem. For example, the factory spindle that came with my MDX-540 mill had a TIR of about 6  $\mu\text{m}$ ; using a hook spanner to adjust the tension of the internal spring by one tenth of a turn reduced the value to 1  $\mu\text{m}$ . In more complex cases, switching around or rearranging the existing bearings inside the spindle, or replacing them with new ones, will often do the trick.

**Tip:** try to quickly repeat the  $R_{\text{collet}}$  measurement after every tool change, even if you are not doing high-precision work. Trapped dirt will affect the concentricity of the tool holder, and even a tiny difference can easily reduce tool life by 50%, and increase cutting noise by more than that.

### 2.3.2. Axial alignment of the tool

Spindle eccentricity aside, it is also useful to verify that the tool is truly perpendicular to the X-Y plane; if that's not the case, for example because one of the screws that attach the spindle to the rest of the mill is not tightened to the same torque, you may see somewhat perplexing dimensional errors in machined parts.

The test you should perform is exceedingly simple: you need to mount a cutter that offers at least around 2-3 cm of clearance between the collet and the flutes; programmatically lower the spindle by 2 cm or so; and set up the dial indicator as shown on the previous drawing, in section 2.3.1. When done, gently rotate the tool to find the mid-point of its TIR, and then programmatically move the spindle up by about 1 cm. The same procedure should be repeated after repositioning the dial indicator to make contact with the side of the tool (rather than the front).

If everything is fine, you should see no appreciable change in the values shown by the dial indicator; a few microns may be fine, but if the difference is getting close to 0.01 mm, you should definitely investigate. The issue is almost always trivial to fix: you may need to loosen the screws that hold the spindle in place, and perhaps insert a shim made out of aluminum foil on the offending side to straighten it out.

**Caution:** before operating the mill, be sure to read the safety tips provided by the manufacturer, as well as the advice included in [section 7](#) of this guide. Small CNC machines are not particularly deadly power tools, but they are still power tools - and it's your responsibility to use them safely.

### 2.3.3. Spindle vibration

The spindle assembly is typically fairly heavy, and under normal operating conditions, will be rotating rapidly. At these speeds, any poorly balanced rotating part, any malfunctioning ball bearing, and any damaged transmission belt may easily introduce significant vibration - and that vibration will inevitably propagate to the end mill or the

workpiece.

It is difficult to accurately measure high-frequency vibration without the help of specialized tools, but this shouldn't stop you from performing two rudimentary checks. Try this:

- Attach your dial indicator to the table or other part of the mill frame, and let the tip touch any non-rotating part of the spindle. Run the spindle at several different speeds, and observe the reading; oscillation of the indicator hand, if any, should stay well under 1 µm.
- Remove the tool, grab a long screwdriver, and carefully touch any non-rotating part of the spindle with its tip. Be vigilant, and keep your body and the screwdriver itself well clear of any moving parts. When the tool is touching the spindle, you shouldn't be able to hear any ringing noise, or sense any appreciable vibration in the palm of your hand.

If any of these tests reveals excessive vibration, the first thing to do is disenage the motor from the spindle (there's usually a belt or some sort of a clutch involved); if the problem doesn't go away, you know that the problem is with the motor itself, in which case, it may be useful to have it serviced or replaced. If the spindle is to blame, replacing the internal bearings would be the obvious next step.

Whatever the cause is, fixes shouldn't be too expensive, but pinpointing the issue may take a while.

#### 2.3.4. Repeat accuracy

---

Repeat accuracy is the single most important factor limiting the precision of the parts you can make. Even if you are not planning to machine anything particularly intricate, this parameter is still worth checking: if it's alarmingly poor, it may be indicative of a problem with the mill.

To estimate the accuracy of the machine, brace the tip of the dial indicator against the side of the spindle assembly, and then programmatically move the spindle away in the X axis, in 0.01 mm increments (or whatever the nearest multiple of your mill's mechanical resolution is supposed to be). After about 5-10 steps or so, reverse the direction, and gradually move back to the starting point. Here's what to look for:

- Every step should move the needle of the dial indicator by the same amount, ideally within 1 µm, and definitely within 0.01 mm. Jerky stepping may be due the increments not being a multiple of the actual hardware resolution of the mill, or due to wiring problems in the stepper motor.
- The actual travel distance, as measured by the dial indicator, should match the expected distance closely. If there's any substantial error that seem to be proportional to how far you are moving, your CNC software should let you compensate.
- The locations of the steps when moving away from the dial indicator should be the same as when moving back; and the spindle should return to the same starting position at the end of the test. Any observable offset is probably due to backlash - which should be in the micron range. There may be some limited ability to compensate for this in software (for example, your CNC application may offer low-level motor calibration settings), but it's not a silver bullet.

Of course, try to repeat this procedure for all axes. If in any of them, repeat accuracy is worse than 0.01 mm, it may be good to talk to the manufacturer.

#### 2.3.5. Deflection under load

---

We're almost done! The last parameter of note is the loss of accuracy you can expect if the mill is braking or accelerating rapidly, or aggressively plowing through a difficult workpiece. The value depends on the rigidity of mill frame, and the type and quality of its linear motion systems. If it's poor, there is no reason to despair - but it means that you may have to slow down when doing precision work.

The test here is extremely simple: with the mill on but the spindle turned off, brace the dial indicator against the side of the cutting head, and then use your hand to gently press the spindle from the other side - along the tested axis. Don't overdo it: the goal is to exert may be 20-50 g of force, and not to overcome the holding torque of the motors.

In a quality mill, the momentary deflection should stay under 5 µm or so - but up to 0.02 mm is something you can live with.

## 2.4. Ongoing maintenance

---

Once you know that your machine is behaving correctly, there isn't that much that needs to be done on an ongoing basis: it's a pretty sturdy piece of machinery, and it's usually not subject to heavy wear. Consult the manual for manufacturer-specific advice - but in most cases, the rules are pretty simple:

- Always keep the collets, cutters, and the spindle taper squeaky clean and lightly lubricated. Get rid of any thick, factory-applied grease (use WD-40 or naphtha to do that); carefully remove all cutting residues (use a [small brass brush](#) for any stubborn bits); and apply WD-40 or a similar low-viscosity oil to protect all these components against corrosion - especially after handling them with bare hands. Sweat residue can quickly lead to corrosion!
- Thoroughly vacuum the mill after every job. Cuttings, if allowed to pile on, may eventually make their way into the sensitive, normally shielded parts of the drivetrain - or more prosaically, may simply obstruct the movement of the table. Do not use compressed air for cleaning, as this can push dirt deeper into the machine, or just spread it around your workshop unnecessarily.
- Every few hundred hours of machining (1-2 years' worth of weekend projects), wipe old grease and dirt off of ball screws and linear guides, and apply a fresh coat. Medium-viscosity [lithium grease](#) is typically the right choice - but check the manual carefully, and whatever you do, avoid mixing lubricants of different kinds.
- After several thousand hours of use (5+ years), it may be time to inspect the spindle assembly for any obvious wear, loss of concentricity, noise, vibration, or elevated operating temperature - and service it as necessary. If you do it yourself, the replacement of spindle bearings should cost between \$40 and \$300, depending on the design.

Some of the entry-level mills may be using low-cost brushed motors to power the spindle; such motors are a consumable, and may require a replacement after anywhere from 100 to 2,000 hours - but typically don't cost much. Higher end machines usually rely on brushless motors that should last a decade or more.

Linear drivetrain motors, bearings, and so on are typically not subject to substantial wear when doing lightweight hobbyist work; if properly maintained, they should last pretty much forever. Insufficient lubrication or contamination with abrasive materials (ferrous metals, glass, etc) are about the only things to watch out for.

## 2.5. Cutter management

---

If you are planning to do high-precision work, or simply wish to ensure high-quality surface finish when working with organic shapes, it's useful to measure and document the diameter of every new end mill in your collection. Although the tools are usually manufactured with micron-level accuracy, the specifications can sometimes be wrong, and on top of that, manufacturing variations may occur from batch to batch - for example, due to changes in the thickness of applied coatings or the gradual wear of the grinding tool. Case in point: some of my Harvey's 0.04 inch cutters actually measure around 1.022 mm, rather than the expected 1.016 mm. At these scales, such differences can bite.

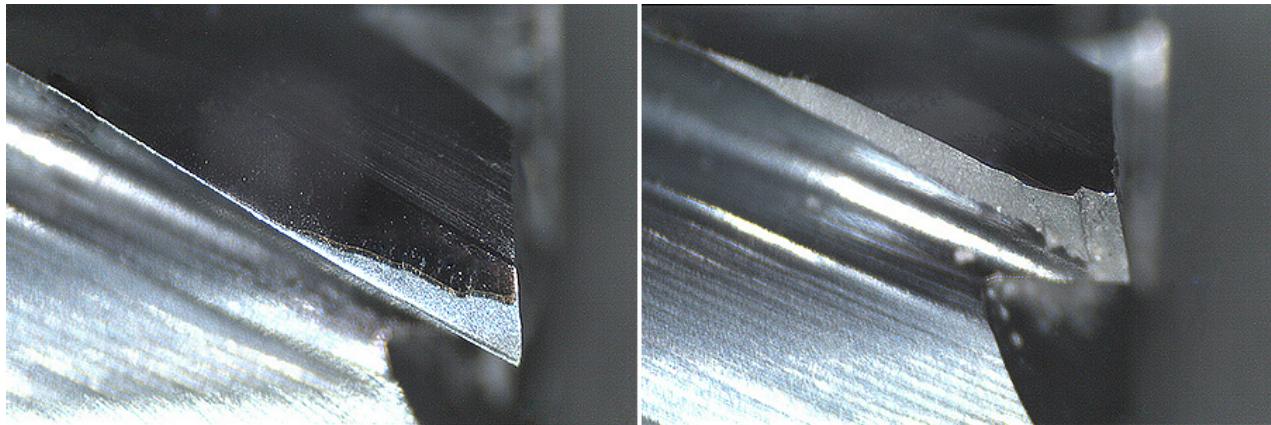
To perform the measurements, you will need an accurate micrometer. This tool, along with quality calipers, is one of the most important investments you will make, so don't fret: \$50 will get you a [decent no-name brand](#), while \$140 is enough for [Mitutoyo](#). For two- or four-flute tools, the idea is to gently tighten the jaws of the micrometer around the flutes, while simultaneously rotating the cutter (in the direction opposite to its normal operation) to find the maximum diameter. You need to stop as soon as you feel any resistance, to avoid breaking the tool or gouging the jaws; practice on larger, sturdy end mills before moving on to sub-mm ones.

Three-flute tools are a bit harder to deal with. If the flutes are long enough, you may be able to grip the cutter so that one face of the micrometer is touching the peak points of two flutes, and the opposite face is touching the remaining one. That said, with stub-length tools, you may be essentially out of luck; doing a careful test cut and measuring the result may be the way to go (you need to account for TIR and repeat accuracy).

Beyond the initial measurements, it is also a good habit to re-check your tools after every 10 hours of cutting or so, preferably measuring the diameter near the very tip. When doing heavy cutting, you may see some reduction in tool diameter as the outer edge of the flutes gets a bit more dull; for this reason, I suggest keeping your primary roughing tool separate from the finishing ones.

**Tip:** even when working with plastics, applying several drops of a cutting fluid to the workpiece can improve finish and limit tool wear by keeping the tool cool and helping remove chips more efficiently - give it a try! For lightweight work, sulfur-free oils, such as [Oatey 30200](#), work best. They are also easy to clean up with a detergent and don't interfere with resin casting work.

Cutting fluids or no fluids, it makes sense to examine your tools for damage and excess wear every now and then. You can't trust your naked eye, but a simple [7x magnifier](#), selling for under \$50, should do the trick. If you have a microscope with magnification between 10x and 50x, that's even better. Here are the two most common cutter failure modes that aren't visible with naked eye - significant wear (left) and a chipped flute (right):



Both of these tools will still work, but the one on the left will not hold tight dimensional tolerances in particularly demanding applications; while the one on the right will produce crummy finish in machined parts and will be prone to gumming up.

**Tip:** good bookkeeping is incredibly important in CNC work: computer-aided design and creative chaos simply don't mix. Cultivate good habits starting with end mills: make sure that you have a spreadsheet (or even a flat text file) listing all the tools you have, outlining their geometry, and making note of the measurements you have taken.

Having such a list will not only help you avoid surprises, but will also make it easier to maintain a healthy stock of tools - so that you never have to put a project on hold for two weeks after accidentally breaking your last 0.4 mm end mill.

## 3. Mastering CAD and CAM

By now, you should have a good idea of which mill to choose, where to find the cutters, and how it all fits together... or perhaps you own a 3D printer, and didn't read the previous chapter at all. Either way, the next step is getting comfortable with the software needed to bring your ideas to life. In this section, we'll go over some of the basics, and then proceed with a simple starter project of our own.

### 3.1. Computer-aided design (CAD)

The primary function of CAD software is, quite simply, to let you design 3D parts. Your CAD application may be just about any general-purpose modeling program, such as [Blender](#) - but in the long haul, it makes sense to settle for a purpose-built tool. "Real" mechanical design software offers better control over part accuracy, and comes with powerful data input and analysis tools that streamline engineering work. On the flip side, they usually have less

impressive rendering capabilities - and may come with no support for animation, physics, and other perks taken for granted in general-purpose 3D apps.

For now, though, the distinction between general-purpose modeling tools and CAD isn't that important. You simply need to get comfortable with any 3D design software of your choice - and that takes a bit of work.

### 3.1.1. Wait, but which application to choose?

---

A-ha, that's a good question indeed! In the previous edition of this guide, I tried to give an impartial overview of the market - but in the end, there is plenty of choice, and very few genuinely bad CAD tools. You can just look around, find the one you like and can afford.

If you just want a simple recommendation - and are willing to spend some money on software to begin with - [Rhino 3D](#) is probably the best CAD package that you can get on a hobbyist budget. Students can purchase a fully-featured [edu license](#) for under \$140, so if you are still in the academia, it would be foolish not to go for it. For mere mortals, there is a heftier price tag attached - \$750 - so it's a more difficult call. Still, it's a mature and user-friendly tool that runs well even on low-end systems, and it's just done well - so you probably won't regret it.

Now, if Rhino is priced outside your league, many people in the community are also fond of [Alibre Design PE](#), which sells for about \$100. Or, if you prefer not to spend any money at all, and can live with a somewhat clunky app, then [FreeCAD](#) looks fairly OK for simpler work.

What else? Several accomplished DIYers use general-purpose 3D modeling tools such as [Sketchup](#) or [Blender](#). The free-of-charge general-purpose tools are pretty diverse and mature; the only problem is that they may be less suited for complex work later on. If you are willing to cross that bridge when you come to it, they are definitely worth a try as a starter option.

Last but not least, cash-strapped DIYers may also want to check out one of the "demo" editions of commercial CAD tools. For example, [Creo Elements](#) has a modest limit of 60 parts per document, but otherwise, should do the trick.

### 3.1.2. Some existential CAD advice

---

We should probably start the lesson with a gentle warning: CAD programs tend to have a fairly steep learning curve. This is in part because you are forced to manipulate 3D objects using a 2D input device and a 2D display - and it takes a while to master that skill. The other problem is that these applications tend to use unfamiliar UI paradigms and obscure terminology - and even something as simple as right-clicking an object may have an unexpected result. It takes some effort to start using the software in a competent way - and if you're just banking on your innate abilities, you will probably learn to do things exactly the wrong way.

If you want to make real progress, here are some rules to live by:

- **Read the manual and do the tutorials.** Trust me, you are not too good for this - even if you never had to do that with any other software. Set aside 2-3 days to go through all the exercises, and read the help pages. You need to understand what the program has to offer and what terminology it uses; even if you have some experience with 3D modeling software, you will learn a lot.
- **Give your mouse a break.** Try to do as much as possible without clicking around the UI. Learn command names and keyboard shortcuts, and make a habit of entering coordinates and dimensions manually. Find hotkeys for enabling, disabling, and changing grid and object snaps, changing drag modes, constraining movement, zooming and rotating viewports, and so on. This will save you plenty of time.
- **Don't approximate.** Never move objects around to make them just sort-of fit together. Learn to leverage object snaps, or use parametric operators to move, copy, rotate, or replicate objects with precision. If you don't, you will quickly start accumulating subtle geometry errors that will cause many 3D operations to unexpectedly fail later on (e.g., due to self-intersecting curves or imperceptible gaps).
- **Don't believe in magic.** CAD programs often come with advanced features to automatically calculate intersections or unions of 3D solids, place and remove holes, etc - but these functions

aren't perfect, and tend to misbehave at least opportune times. Instead, learn to break down complex problems into simple steps, and operate on intuitive primitives (e.g., planar curves and surfaces) for as long as possible in the editing process. This also makes it easier to revise your designs later on.

Before you start, there's just one more thing to do - you need to customize the program for precision work. Needless to say, the tolerances needed to design a building aren't the same as when designing a gearwheel - and CAD applications are used for both. For now, simply go through the configuration pages, and try to do the following:

- **System units:** set to millimeters. Or don't, but I find inches pretty unwieldy for precision work.
- **Precision:** set precision (or tolerance, or accuracy) to 0.5  $\mu\text{m}$  (0.0005 mm) or so. Display mesh tolerance can be set to about 0.01 mm, simply to speed up rendering.
- **Display grid:** enable if necessary, and set major spacing to 5 mm, and minor spacing to 1 mm. Set grid size to match the work area of your machine.
- **Snaps:** disable object and angle snaps by default (you will be able to turn them on with a hotkey when necessary). Enable linear snaps and set distance to 0.5 mm. If the program supports it, turn on "planar" mode for snaps, where all the object snap points are projected to the construction plane of the current viewport.
- **Viewports:** switch to four-viewport display mode - showing top, front, and right sides of the model, plus a rotatable "perspective" (axonometric) view. Zoom in all viewports so that there are about 2-4 mm between the grid lines displayed on your screen.
- **Shading mode:** configure all viewports to use 3D shading, preferably with translucent surfaces (opacity around 80-90%; "ghosted" preset in Rhino), and with 1 pixel edge thickness. If possible, set a different edge color (e.g., red) for any "naked" edges that do not form a closed solid.

Done? Then let's roll...

### 3.1.3. Drawing and manipulating simple planar shapes

Your first job is to figure out how to draw several simple, two-dimensional shapes on the X-Y construction plane (i.e., using the "top" viewport of your CAD app). Try to sketch all of the following:

- A single, straight line segment,
- A multi-segment zig-zag line (with sharp turns),
- A closed multi-segment line, where the end is joined with the beginning,
- A circle, using three methods: center point and diameter; two boundary points; and three boundary points,
- An arc - by selecting center point, diameter, and start / stop angles,
- A rectangle, using two methods: two corner points; or a center point and one corner,
- A polygon - circumscribed and inscribed, with a specified number of sides.

Practice a bit; perhaps sketch a simplified, boxy outline of a car, complete with wheels.

When you are comfortable with these 2D primitives, it's time for your next exercise: try to draw a smiley face without using your mouse at all. Type in the required commands and specify coordinates by hand; the display grid should be of great help. Oh - for some extra credit, add a hat!

With your drawing in place, it's time to get familiar with several important operators. Figure out how to select one or more objects with your mouse, and then find the commands that perform the following tasks:

- Move selected objects by a specified distance in the current viewport,
- Duplicate ("clone") an object, likewise,

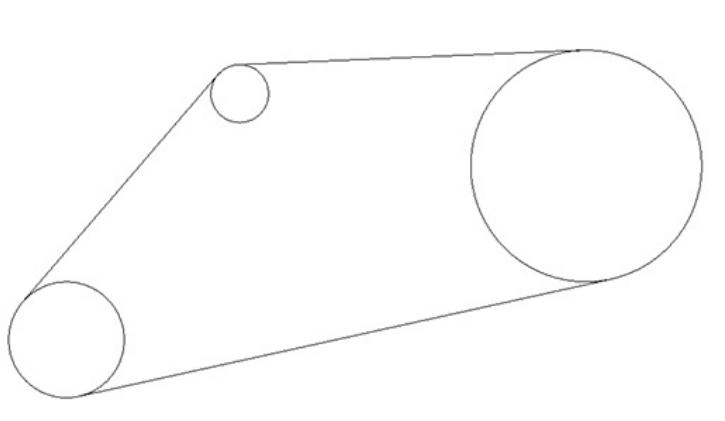
- Rotate object by a specified angle, around a selected point in the viewport,
- Flip (or "mirror") selected objects around a specified axis,
- Scale object proportionately in all directions, by a specified factor,
- Scale object in one direction only,
- Form an array of objects by cloning the selection at a given linear interval,
- Form a radial (polar) array by cloning the selection at an angular interval.

Play around with these operators until you are comfortable with the way they work; pay special attention to scaling and rotation operators, and the way they are affected by the choice of the origin, reference point, and the viewport.

As soon as you are done moving, flipping, and cloning stuff, locate and play with the analytic tools that let you do the following:

- Measure the distance between any two points,
- Measure the angle between two straight lines,
- Calculate the diameter (or radius) of a circle or an arc,
- Find the center point of a circle or an arc,
- Show maximum deviation between two similar curves,
- Compute the surface area of any closed curve.

This is a good opportunity to experiment with object snaps, too: enable them temporarily, and check the various options they offer. In particular, be sure to give tangent snaps a try: draw three random circles, and then try to connect them with tangent lines, like this:



Piece of cake? Thought so! You should be now ready to master several more complicated modeling skills.

**Note:** Somewhat surprisingly, 2D drawing techniques are more important than any 3D sketching tools; in fact, you should resist the temptation to play with 3D primitives at this point. The bulk of mechanical modeling work is almost always done with [spline curves](#), which are later converted into 3D objects with the help of operations such as extrusion, revolution, or lofting. We'll get to these operators soon.

### 3.1.4. The inner workings of splines

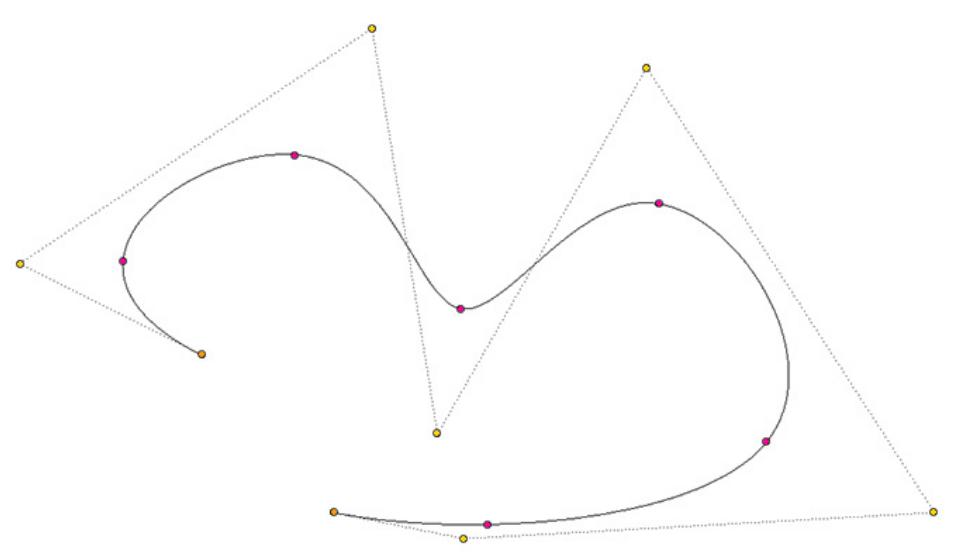
Your CAD application probably stores every curve as a [non-uniform rational B-spline](#). The visual representation of this mathematical model is not very easy to grasp, but getting a hang of it is essential to any sort of serious modeling work.

In essence, every NURBS curve is defined by three parameters:

- **Degree:** this parameter defines how many control points will be simultaneously affecting the trajectory of the curve at any given location. Degree 1 curves allow just two points to have a say, with a linear transition between the starting and the destination point (i.e., a straight line). For degree 2 curves, the shape is influenced by three points, allowing it to have smooth bends. With degree 3, the number grows to four, and so on.
- **Control points:** every curve is accompanied by at least **degree + 1** control points, to which it is progressively attracted to along its length. In degree 1 curves, control points simply lie on the curve (at the vertices). For degrees 2 and above, the points are often positioned outside the curve, forming a sort of an editable, elastic envelope, as shown below.
- **Knots:** these are the intervals along the length of the curve at which the influence of a previous control point ends, and the influence of a new one begins. Think of it as a way to pace the changes: densely packed knots can result in sharper turns.

*Note: it is possible to have multiple subsequent knots at the same location. If the number of knots in a cluster is equal to the degree of the curve, it's called a kink: it effectively forms a barrier that prevents control points on one side to have any influence over the other, and vice versa. This allows the curvature to change abruptly, forming a proper angle. Curves can be losslessly broken into separate segments at such a kink, too.*

An example of a degree 2 spline with no kinks is shown below; control points are marked in yellow, and knots are red:



To practice a bit, try to locate the command that lets you draw a curve of a given degree by specifying subsequent control points. Get a hang of its behavior particularly for curve degrees 2 and 3. There should be also a separate command for creating interpolated curves that go through any number of specified "via" points that you click on, which may be useful for tracing around bitmaps and so forth. Oh, one more thing: using both of these tools, try to create a proper kink!

When you are comfortable with drawing, experiment with the following curve-editing tools:

- Displaying and moving around control points for an existing curve,
- Adding and removing control points, inserting a kink,
- Non-uniform refitting, to simplify the curve or remove kinks while maintaining its approximate shape,
- Soft-editing a curve by dragging a selected point on its surface,
- Displaying curvature and continuity information - can you spot any existing kinks?

Last but not least, be sure to familiarize yourself with the "explode" operator that splits curves at existing kinks (see what it does to a rectangle, circle, etc); and the "join" tool that merges adjacent curve endpoints to form a curve with kinks.

### 3.1.5. Curve trimming

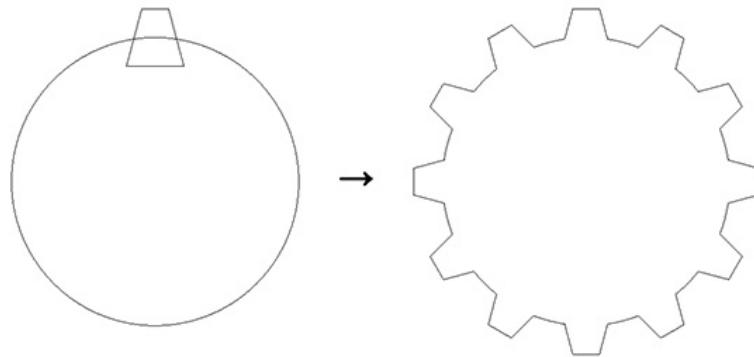
As should be apparent by now, NURBS curves can be separated and joined at kinks in a completely lossless manner: the underlying coefficients do not have to change, and there is no gradual reduction of accuracy if you repeat these operations 100 or 1,000 times.

Unfortunately, in many cases, you will need to truncate curves at locations that do not correspond to any existing kink. In these situations, the CAD application will be usually forced to create a new curve with a different set of control points in the vicinity of the cut. This fitting process usually works very well, and the resulting deviation should be much smaller than your configured tolerance, but you should be aware that the operation does not come completely free.

With that warning in mind, find the command that let you split a curve at a point where it intersects with another one, and experiment with it. For degrees higher than 1, you should be able to see the effect it has on the set of control points for the curve - and if you ask the program to calculate the deviation between the input and the output curve, it will probably give you a tiny but non-zero result. Examine the deviations for a couple of input curves to see what to expect in normal work.

In addition to the curve splitting tool, there may be also a second operator that works roughly the same way, but lets you immediately delete (trim) the unwanted parts. It may sometimes save you a click, so if possible, get familiar with it.

In any case, it's time for a brief exercise. Remember the command for drawing radial arrays? Try to combine that with splitting and trimming to arrive at a result such as this:



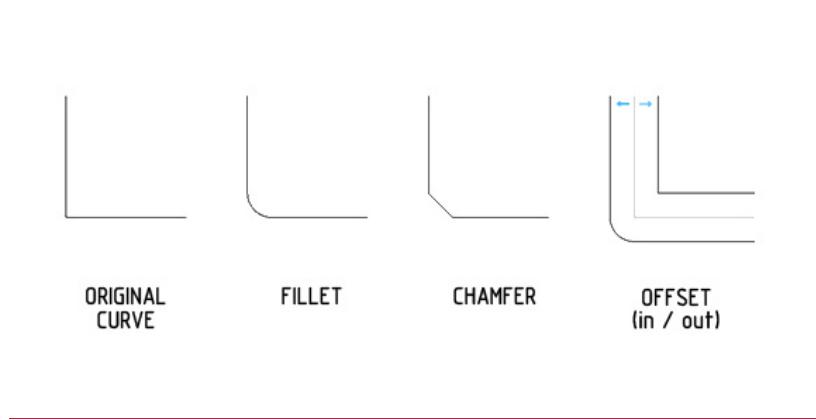
Nice, eh? It's not quite a proper gearwheel, but we're pretty close.

**Note:** as mentioned earlier, it is always preferable to use the simplest editing tools that still do the job. In particular, you should always favor the join, explode, refit, split, and trim operators over any features that, for example, automatically compute unions, differences, or intersections of 3D objects.

Why? Well, these advanced tools quickly go bonkers if you have something as simple as two closely touching, parallel walls - not to mention accidental self-intersections, non-watertight curves or solids, etc. It's one thing if they just fail in an obvious way... but if the resulting error is subtle and goes unnoticed, you may end up having to redo a good chunk of your work later on.

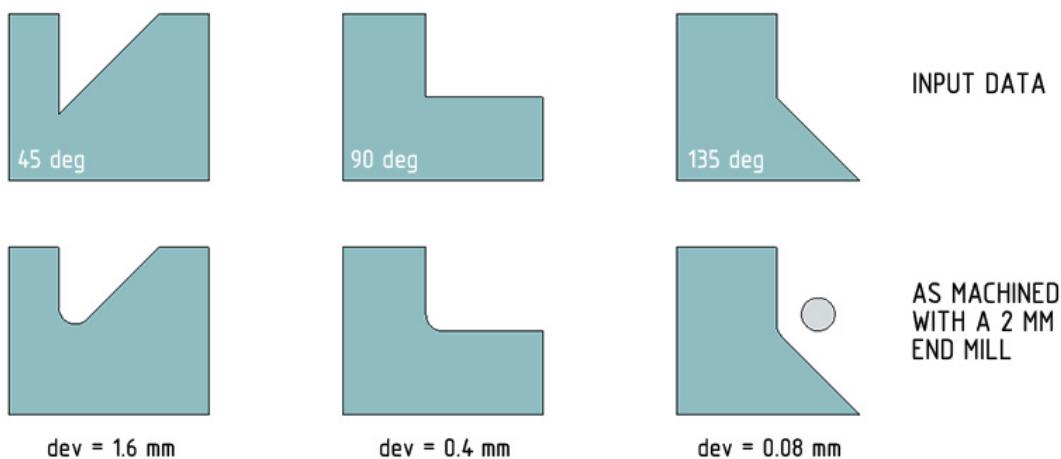
### 3.1.6. Other curve tools

We're almost done with curves - hooray! But before we go, there are three other, immensely useful editing tools that you should probably learn about:



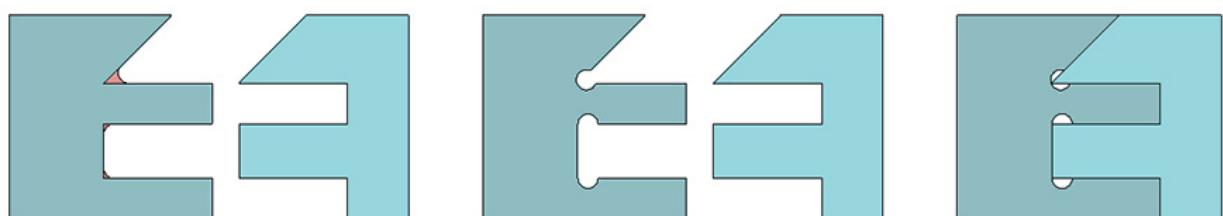
The fillet operator is applied to any angle (kink) on a curve, and replaces it with a specified radius. The trick is often used for engineering purposes: fillets serve as a stress relief in sharp corners and drastically improve their strength. But even more interestingly, filleting lets you capture the real-world result of machining a corner with an end mill of a given size - which is important for certain types of snap fits.

This illustration shows the use of filleting to determine the actual shape of a machined part:



**Note:** Of course, it's not that you can't make a competent acute angle with a CNC mill. For example, refinishing the one on the left with a 0.4 mm cutter would result in a deviation of barely 0.3 mm. Still, in many cases, it's more convenient to account for such intrinsic fillets in your designs, than it is to refinish the workpiece with a separate tool.

Oh, one more thing: in situations where you don't control the shape of the mating part (lighter blue in the picture below), you can always resort to bone fillets, too:



Chamfering is a similar operation, but it replaces the selected kink with a straight segment that starts at a specified distance from the vertex. Compared to fillets, chamfering may have a more desirable appearance, depending on the effect you are aiming for - and produces a much simpler 3D mesh.

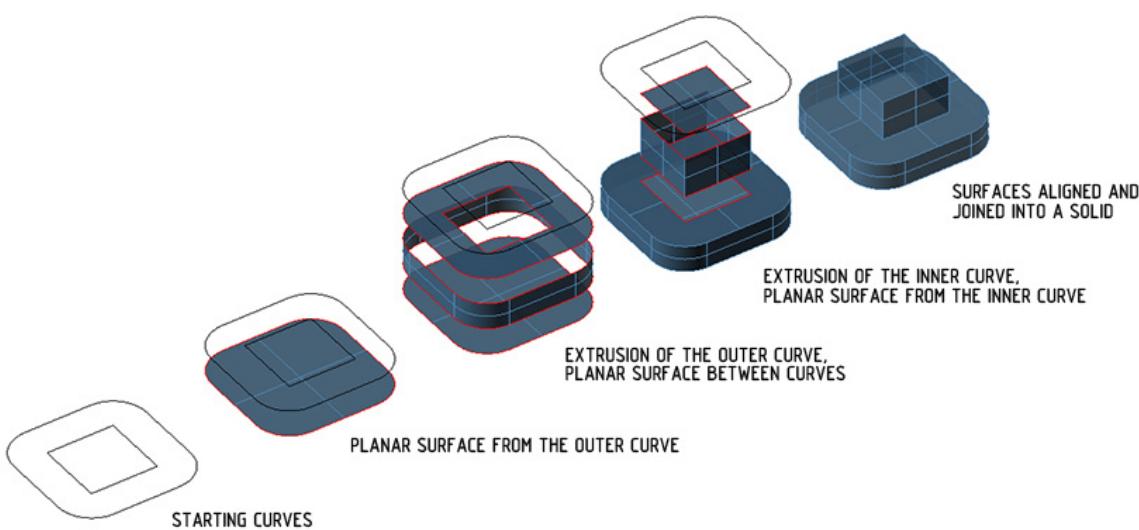
The last operator, offsetting, offers an interesting way to resize any shape: it creates a derived curve that consists of points at a particular, constant distance from the original geometry. It is quite different from traditional scaling, in the same way that gaining weight is different from growing bigger - and it comes quite handy for generating walls, undersizing or oversizing parts, and so forth.

Well, that's really it! You may also want to explore tools that let you extend or close curves, or blend them with more fine-grained controls - but for most intents and purposes, we are ready to talk about something a bit cooler than that.

### 3.1.7. Creating simple surfaces

Let's start building three-dimensional solids. It's simple: draw a closed curve, and then find the command that creates a planar surface from it. Examine the result, then check out what happens if you select two curves, one within another. What about separate or intersecting input curves? Examine failure modes of this operator, too: what if one of the input curves is at a slightly different Z level? What if the curve is open or self-intersects?

Of course, the planar surface that you have just created is still, well, pretty flat. The next important operator to play with is extrusion: select your curve, and using the side view, select the extrusion height. Disable the creation of top and bottom "caps": we just need the side walls. Try to use the join operator to merge the flat base surface with the extruded wall. If it worked, you have pretty much mastered 3D work. Lets try a slightly more complicated exercise:



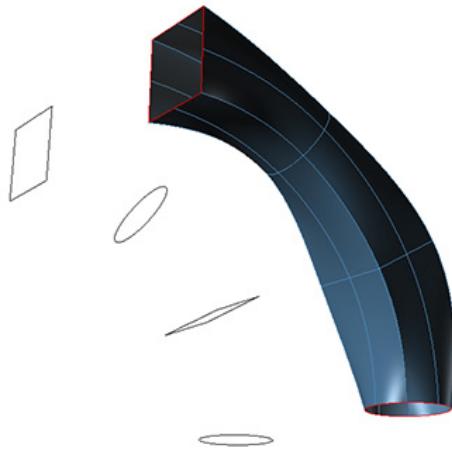
That's pretty easy, eh? In fact, it's a lot simpler to construct shapes this way than it is to start with 3D primitives, and clumsily piece them together. But don't let it go to your head. Spend a few more minutes playing with extrusion: see what it does to open or self-intersecting curves, to multiple curves that overlap or aren't at the same Z level, and so on.

You should also take this opportunity to experiment with the explode, split, and trim operators that you remember from your curve editing days. Draw two solids and make them overlap, then see if you can use the trim operator to remove all the overlapping internal sections, and join the resulting outer shells into a new, closed shape. Can you do the same to "subtract" one solid from another, or only keep the intersection? Find the analytic tool that lets you calculate the volume of each of the solids, and use it to double-check that your results check out. Oh, one more thing: can you trim surfaces with curves? Why not give it a try!

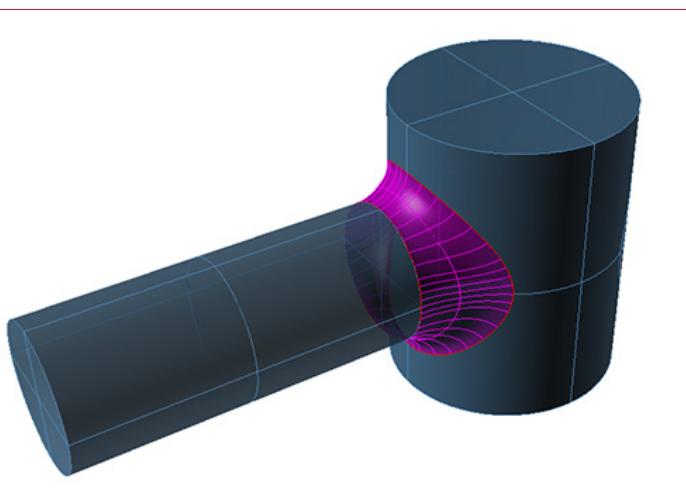
These tools aside, there are several other advanced surface editing operators that you should be aware of; they are particularly important if you want to create organic shapes without having to think too much:

- **Curve revolution:** this operator lets you draw a surface by taking any curve, and rotating it around a specified axis, and letting it leave a trail in the 3D space. This is the method of choice for creating many simple shapes with axial symmetry - bottles, vases, cans, chess pieces, rings. In fact, try to create each one of these.
- **Lofting:** this mechanism lets you "drape" a surface through a series of curve outlines arranged in the

3D space. Think of it as the opposite of slicing. It comes handy in a variety of situations, including creating draft angles for molds, or adding a handle to a cup. Play with it a bit - it's fun:



- **Sweeping:** this tool lets you select a curve and drag it along a specified rail, generating a surface in the process. It can be seen as a generalized form of curve revolution (in which the rail is always circular). A more advanced version of this operator lets you use multiple rails, which allow variable scaling of the profile curve; or multiple profiles, thus also generalizing the loft operator.
- **Blending, filleting, chamfering:** equivalents of the curve editing tools that let you smoothen the edges of your solids. Blending is particularly useful for creating a smooth, variable-curvature joint:



The list of useful operators goes on - for example, it's sometimes useful to extract edges from surfaces or solids, to place or remove holes, or to orient something on an oddly-shaped surface - but you can discover these on your own. For now, let's just have a quick look at how to manage your work.

### 3.1.8. Working with multiple objects

It's one thing to sketch a nice 3D box - but any medium-scale project may easily consist of several hundred of such solids, often meshing closely and stacked in creative ways. If you don't manage your virtual workspace well, you will get overwhelmed, make mistakes, or both.

The document management tools at your disposal will vary from one program to another, but you should familiarize yourself with at least the following:

- **Grouping.** This function lets you combine any number of objects into a single logical component that can be moved around and copied as a whole. Unlike trimming and joining, this operator does not require the objects to form a single solid - in fact, you can group completely disjoint things, such as curves, dimensions, and even light sources for the rendered scene. When necessary, the group can

be opened, and its contents can be edited separately.

This tool is very useful for conveniently creating serviceable parts from simpler, reusable solids.

**Note:** some programs go beyond simple grouping, and let you instantiate objects loaded from a separate document. These instances are automatically updated to reflect any changes to the "master" document. I find this to be an overkill in my work, but your mileage may vary.

- **Assigning to layers.** Layers offer a simple way to centrally manage certain properties of objects, such as visibility or color, without merging them into a single blob. All objects or groups that comprise a named layer can be still accessed individually - although you can also select all members of a layer if so desired.

You should use layers to keep track of all instances of any given part, or to denote other important information - for example, that a particular object represents a premade part not meant to be machined, or that it's only a visual aid.

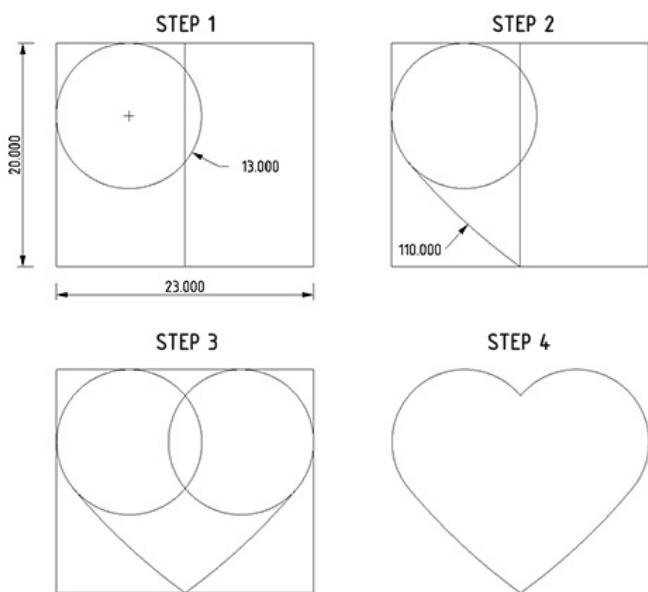
- **Locking and hiding.** Objects or layers that get in the way of editing may be locked. This keeps them at least somewhat visible, but prevents any interaction: you can't accidentally select them or move them around. Hiding is a more radical alternative: it completely removes the affected items from view.

Locking and hiding is typically best applied to layers, as it usually takes just a single click to toggle this parameter in the layer manager. For individual objects, hiding is useful predominantly for items that you are not planning to access again: for example, any construction curves that were already used to create all the solids you needed, but that are worth keeping around just in case you need to redo something later on. To get them back, you will need to activate a special mode that hides all the visible objects, shows all the hidden ones, and lets you select the ones you want to bring back to life.

As soon as you are reasonably comfortable with object management, we can actually make a simple mold!

### 3.1.9. Practice time: let's show some love!

Okay, it's time for some fun. Let's start by trying to draw a heart:

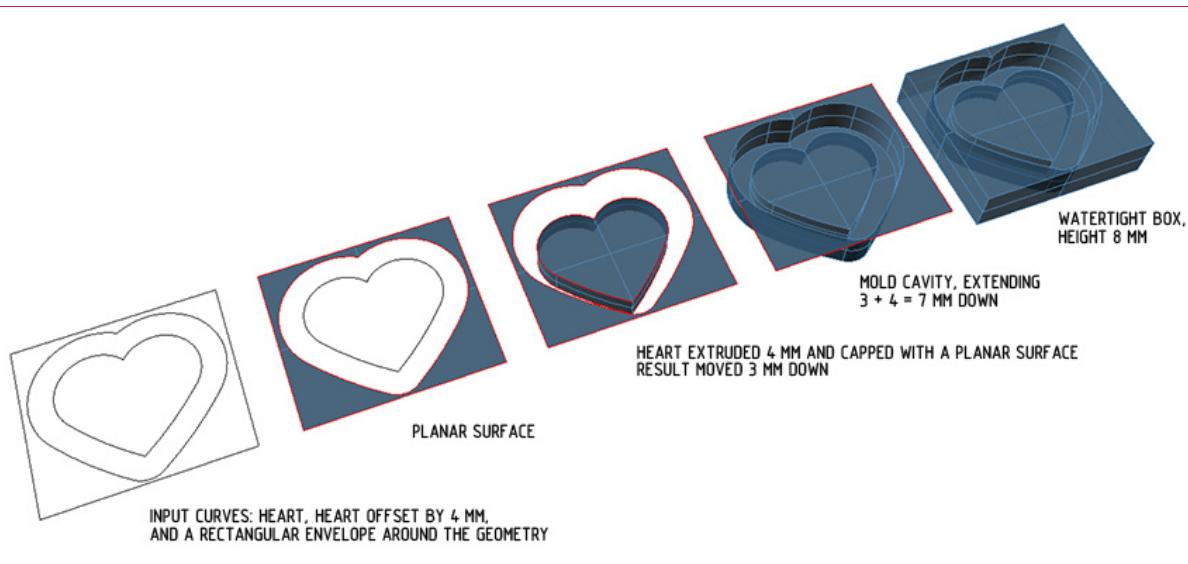


Here's what's going on in that illustration:

- We draw a helper rectangle as a guide for the overall size of the part. This one is 23 mm wide and 20 mm tall. We then use object snaps to draw a vertical line through the middle of the part, which will be our axis of symmetry.

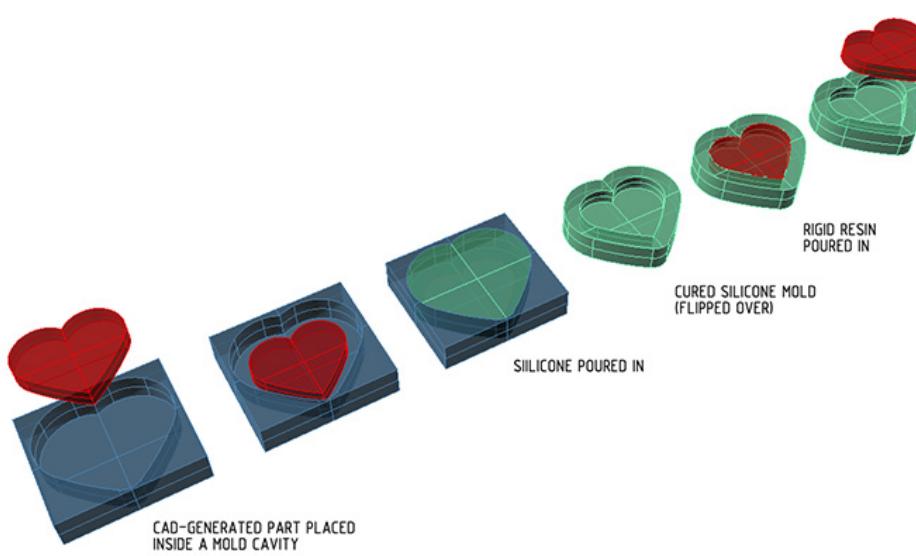
- Once the guides are sketched out, we add a 13 mm circle (slightly larger than half the rectangle), and align it with the edges of the box. You can use grid snaps for that.
- We draw an arc using object snaps. The beginning of the arc is tangent to the existing circle, the end is at the bottom of the center guide, and the diameter is manually entered as about 110 mm.
- We mirror the circle and the arc around the axis of symmetry. At this point, the symmetry guide is no longer necessary, so we delete it.
- We use the trim operator to remove the internal parts of the heart, and then delete the outer rectangle.

Voila! Easy, right? Now, let's join all the curve segments, and turn the whole thing into a mold:



See if you can figure out all the steps on your own. The final result should be a watertight box with dimensions around 33 x 30 x 8 cm. Its bottom left corner should be at coordinates (0, 0), and the top surface should be at 0 mm, likewise. The mold cavity should have a 4 mm clearance around the heart, and should be 7 mm deep. The heart itself should be resting at the bottom of the cavity, and should be 4 mm high (its top surface should be at -3 mm).

If everything checks out - well, the good news is that you have yourself a master mold (aka a *pattern*). Now, we cut some corners with the moldmaking process, given the simplicity of this project, but this is how the entire thing usually plays out:



Be sure to save that project - we'll need this file soon.

## 3.2. Computer-aided manufacturing (CAM)

CAM software reads the geometry created with your favorite modeling application, and turns it into toolpaths that can be sent to a milling machine, a 3D printer, or some other automated tool. All the computer-controlled manufacturing technologies use a common set of underlying concepts and have a comparable degree of complexity, but there are many details that remain specific to a particular tool. Since we have to choose one way or the other, the rest of this chapter will focus on the software designed for CNC mills.

### 3.2.1. Shopping for the right application

When I first published the original version of this guide, the makers of CAM applications catered almost exclusively to commercial users. The software was ridiculously overpriced, shipped with archaic UIs, and always included a collection of mind-boggling bugs. Thankfully, the emergence of low-cost mills and serious hobbyists is slowly changing that. Still, there is no single package that would be a sure bet for all users, and you need to understand what sets them apart. Here's a quick list of the things that matter the most for everyday CNC work.

#### 3.2.1.1. Support for fourth axis

If you own a computer-controlled rotary axis, or plan on getting one, you should figure out which 4-axis machining modes are supported by the program - and decide if they are worth the extra price. The three fundamental choices are:

- **Indexed cutting (XYZ):** in this mode, every milling job is handled using traditional, planar three-axis movements. The only new trick is that the program will automatically rotate the workpiece (and the CAD model itself) in between jobs, letting you machine the part from multiple sides with no extra leg work.

This rudimentary approach is probably the most useful one, particularly for engineering work. It works great on geometries that have a small number of distinct, planar faces - for example, a two-sided PCB, a nut, or a rectangular enclosure.

Interestingly, indexed cutting is actually fairly easy to simulate by creating several CAD files with the same part rotated by the desired degree, generating toolpaths for each of these files, and then sending them to the mill and rotating the part in between. It takes a bit more time, and there's a higher risk of human error - but it's up to you to decide if the extra convenience is worth spending several hundred bucks extra on your CAM tool.

- **Lathe-type machining (AXZ):** in this mode, the Y axis is kept in a fixed position, so that the tool hovers directly over the axis of rotation, or stays at a predefined offset. The workpiece is turning continuously as the milling head is moving up, down, and to the sides. If you haven't seen a lathe in action, [this video](#) should clarify what's going on.

Such an approach allows efficient reproduction of smooth, cylindrical shapes that would require indexed cuts from 6 or more sides to approximate the shape by traditional means. Examples may include threaded shafts, worm gears, soda bottles, or rings.

Unlike the previous mode, this one can't be trivially approximated in a program that doesn't support it natively; but it's also not nearly as essential, not unless you're planning to work on jewelry or other artistic stuff.

- **Fully simultaneous four axis machining (AXYZ):** in this mode, the machine is free to move in all four axes while cutting, and the application analyzes the geometry to find the right balance between lathe-like and planar movements.

This may result in more efficient machining of complex surfaces with variable curvature or hard-to-reach undercuts - but it's seldom of any use to begin with, and it can be approximated by alternating between the two techniques mentioned earlier; the extra cost is difficult to justify in hobbyist work.

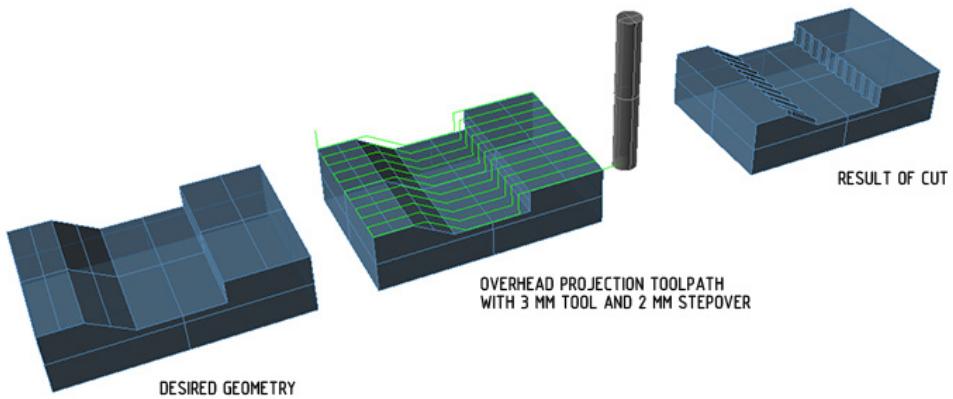
**What to buy:** your call. If indexed cutting is all you need, you may be able to live with lower-cost three-axis CAM.

### 3.2.1.2. 3D milling strategies

Vendors of CAM software tend to be pretty creative about the range of specialized and obscure machining modes that they offer, but there are only three options that really matter; the rest is either redundant, or doesn't work well in complex molds.

Here are the three methods in question; the first two really must be supported by the application you want to go with:

- **Overhead projection:** this method amounts to projecting a simple surface-filling pattern onto the workpiece, from the top. In the most basic case, the pattern may be just a bunch of evenly spaced parallel lines, connected together on alternating ends. The tool then follows these lines in the X-Y plane (X-A for lathe-style machining), and is lowered or raised as necessary to reproduce the shape of the part:



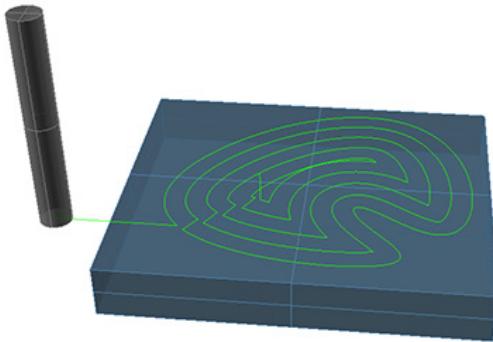
Such a computationally inexpensive approach is particularly efficient when machining flat regions or very gentle slopes with a flat tip cutter; in this case, the spacing between projected lines (*stepover*) may be comparable to the diameter of the tool, and still produce excellent surface finish.

Alas, the strategy is a bit less suitable as soon as the cutter gets near any vertical surfaces. If the wall is in a direction perpendicular to the toolpath, the process will leave scallops of uncut material, as shown above; the height of the scallop is equal to  $\text{cutter\_radius} - \sqrt{4 * \text{cutter\_radius}^2 - \text{stepover}^2} / 2$ . Walls that happen to be parallel to the direction of the cut will acquire a better finish, but the process may leave a layer of uncut material, up to *stepover* thick. Walls at other angles will get a combination of both artifacts.

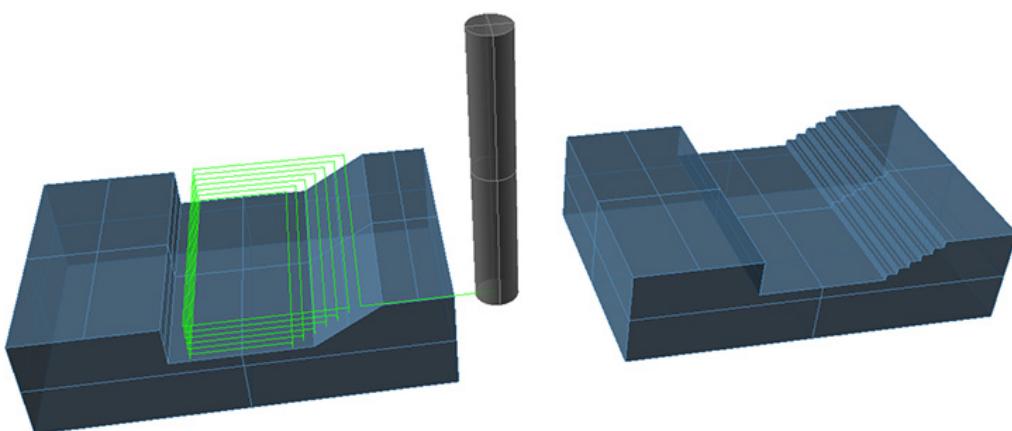
**Note:** both of these issues can be remedied to some extent without switching to a more advanced machining strategy. Scallop height can be managed with modest changes to *stepover* - for example, when using a 3 mm tool, changing *stepover* from 2 mm to 0.5 mm reduces scallop from 0.38 mm to 0.02 mm - that's a 20-fold improvement at the expense of slowing down by a factor of four.

The other problem - with the accuracy of parallel walls - can be fixed by doing a second pass using lines projected at a different angle, so that the cut is no longer parallel in the previously affected area. Of course, the situation gets a lot trickier if you have walls at many different angles, or worse yet, any arcs.

Good CAM software be able to not only generate linear projections, but also "offset" patterns that are constructed from an arbitrary outline provided by the user. This has some important benefits that we'll talk about soon; the end result looks kind of like this:



- **Z axis slicing:** also known as waterline cutting, this machining strategy starts by creating a stack of horizontal surfaces, evenly spaced in the Z axis. Each of these surfaces intersects the CAD model, and every such intersection results in a number of closed, two-dimensional outlines of the part. These outlines can be then offset for the size of the tool, checked for collisions, connected together, and used as a toolpath:



This mode allows for faithful and highly efficient reproduction of intricate vertical features, such as spur gears. On the flip side, it does not do anything useful for horizontal surfaces; and produces staircase patterns on sloped surfaces, unless the spacing of Z levels is unreasonably tight - as seen above.

On top of the basic mode with a constant Z step, some of the more expensive CAM programs allow you to automatically detect and optimize the spacing of Z levels based on the geometry of the part. That sounds nice in principle, but I could never get this feature to work in a sensible and consistent way in any of the programs I have tried.

- **3D draping:** this method generates a toolpath that always maintains a particular distance between cuts, or keeps a particular scallop size, regardless of the shape of the machined part; this is somewhat akin to printing linear toolpaths on a piece of fabric, and then laying it over the geometry to be reproduced.

This strategy can be a significant time saver when working with organic shapes using ball tip tools, but it is not commonly useful in everyday mechanical work.

**What to buy:** the two first strategies must be supported unconditionally, and should come with no strings attached: you should have full control over how they are used and when. The third one is a plus. Advanced features of these modes, and other toolpath generation methods, matter a lot less.

### 3.2.1.3. Region selection capabilities

Efficient machining of complex models requires the ability to decide which tool and which strategy will be applied to various regions of the model - and in what order. To facilitate this process, your CAM application should let you define a freeform region for every machining operation, either by selecting an existing boundary curve imported alongside with the 3D model - or by drawing something by hand. It should let you exclude regions within an existing selection by picking a second curve inside the first one, too.

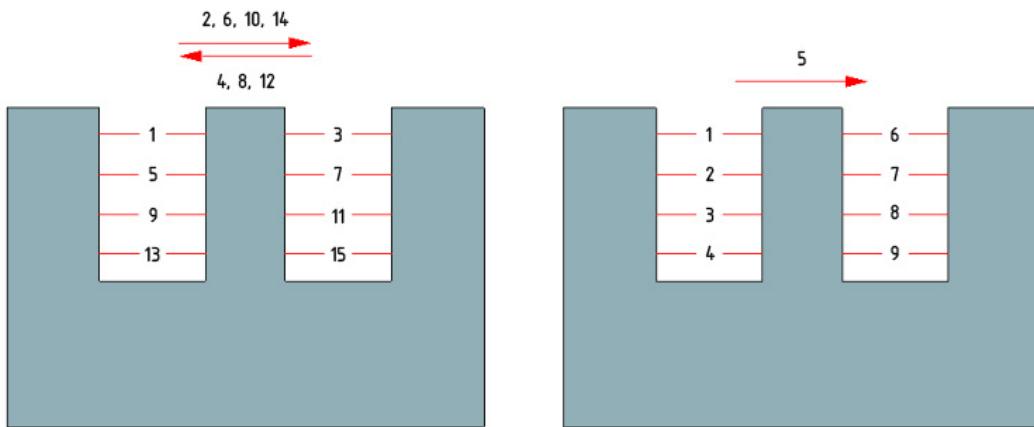
These capabilities are pretty much all that matters. You may notice that certain programs have the ability to automatically generate selections based on some property of the model - for example, to only machine flat surfaces, vertical walls, or something in between. These features are sometimes nice, but are usually more limited or less reliable than what you can do in CAD - so don't pay any special attention to this.

What else? Oh - several programs offer "residual machining" of regions left over after the previous operation with a larger tool. This sounds great in theory, but is typically both fairly coarse and extremely slow; I tried two applications that supported this feature, and it misbehaved or ran out of memory more often than it proved to be useful.

*What to buy:* curve-based selection is a must. Pay little or no attention to the rest.

#### 3.2.1.4. Cut optimization

Some CAM applications do not put any special thought into organizing the toolpath segments that are spewed out by the underlying geometry-analyzing algorithm. Depending on your model, the result of that can be pretty inefficient; for example, a waterline cut that sequentially machines every outline on every layer before moving further down will produce a total of 15 operations when machining these two separate holes (left):



The image on the right shows a more sensible ordering: instead of going back and forth between the machined regions, finish one first, and then move to the other. The result is 9 operations, consisting mostly of short movements. Quite a few programs are capable of such common-sense optimizations - but not all.

*What to buy:* it is useful to shop for CAM applications that offer cut optimization, especially for waterline machining. It's not an essential feature, but it's a major time saver. If in doubt, consult the manuals or ask the vendor before buying.

#### 3.2.1.5. Arc interpolation

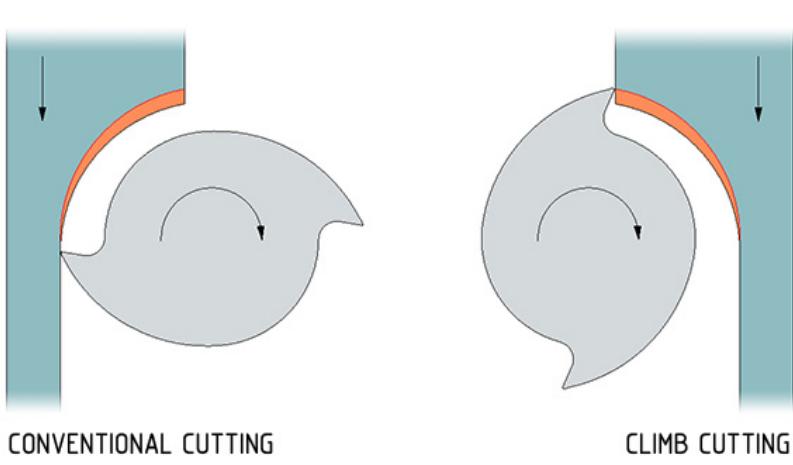
G-code - the language supported by most of the CNC machines on the market - has several commands that can be used to execute circular, spiral, or helical movements of the tool. While support for these opcodes is not required, quite a few embedded controllers recognize them, and translate them to motion optimized for the hardware they are hooked up to.

CAM programs, however, usually convert input files to [polygon meshes](#), and generate toolpaths where arcs are broken into thousands of small, linear movements. This practice is not just a waste of electrons, but may result in lower machining speeds, because it hinders the ability for the controller to plan ahead and pick the best acceleration and braking strategy. Perhaps more annoying is that if the pitch of the generated 3D mesh doesn't overlap with the hardware resolution of the motors in your CNC mill, needless vibration may be introduced, too.

**What to buy:** If your machine supports arc interpolation, it is good for the CAM program to know how to generate G-code that leverages that. It's not a deal-breaker, but something really nice to have.

### 3.2.1.5. Cut direction control

There are some brief moments in the cutting process where the tool may be plugging head-on into uncut material. Most of the time, though, the tool simply works to widen existing pockets, and engages the material one side at a time. There are two principal configurations possible during that task:



In the conventional (aka *upcut*) mode, contact with the workpiece is made on the left side of the tool, in relation to the direction of movement. Toolpaths generated this way allow the material to be engaged very gradually, starting with a near-zero chip load, and ramping it up. This strategy is less dependent on the rigidity of the tool and the workpiece, and results in superior dimensional accuracy and surface finish with rigid plastics and many other types of easily machinable materials.

In the climb (*downcut*) mode, the orientation of the tool in relation to the workpiece is reversed, and every flute plunges into the material to start at the maximum chip load, and then have it taper off. This approach puts greater strain on the tool - and if the tool or the workpiece deflects, it reduces the dimensional accuracy of the final part. That said, it may offer better cutting speeds or improved surface finish in certain annoying materials - such as metal alloys prone to work hardening, or soft, malleable thermoplastics.

In waterline cutting, maintaining a specific direction of cuts comes naturally, and at no extra cost; in fact, it takes special effort to alternate the type of cut between each layer. Overhead projection strategies are more problematic: if the projected pattern is just a bunch of parallel lines, the only way to achieve consistent direction is to cut the first line, raise the tool, move back to the starting side, lower the tool, and make the next cut - in short, there's a lot of non-cutting movements. And that's the major advantage of "offset", spiral-like projections that we have mentioned earlier on: they permit the direction to be maintained without having to raise the tool.

**What to buy:** you should make sure that the software gives you full control over the direction of cut for every machining strategy of note. Not having this ability will make your life miserable, especially when working with miniature, long-reach tools that flex easily.

### 3.2.1.6. Input and output formats

Last but not least, the application you end up choosing needs to support the file formats that can be written by your CAD package. The most common interchange formats include IGES and STEP (both of which are vendor-agnostic); DXF and DWF (originating with AutoCAD); 3DM (Rhino3D); and STL (3D Systems); support for at least one of these is good news - and the more, the better. NURBS surfaces are supported natively in all of these except for STL and DXF, where approximate tessellation may be required.

Input formats aside, you also need to confirm that the application actually supports your CNC mill, and can generate suitable output files. If your machine speaks G-code, and the application offers the ability to create custom G-code postprocessors, you may be able to find a third-party converter or write your own one with relatively

little effort; otherwise, you need to make sure that the mill is supported out of the box.

### 3.2.1.7. ...and all the things you don't have to worry about

---

Here's a short and incomplete list of things that tend to appear in product specifications and may seem important, but typically don't matter much:

- **2.5D machining strategies:** these features are used to convert bitmaps and other 2D input data into engravings, simple cutouts, etc. They are truly useful only if you don't have a CAD application to begin with, which obviously won't be the case.
- **Support for drilling, tapping, boring, and so on:** when working in plastics, it makes virtually no sense to buy specialized tools and use exotic toolpaths to make holes and perform similar tasks; end mills, operated the usual way, will almost always do.
- **Manual toolpath editing:** this ability may be useful for very simple models, where the toolpaths are trivial, and fixing them by hand may be a convenient alternative to reworking the model. Toolpaths in complex molds are nearly impenetrable, though.
- **Visualization support:** the ability to simulate the machining process is a nice addition, especially when making your baby steps with CAM, or when dealing with buggy software - but ultimately, it's not worth the extra price. Once you work out a sensible machining workflow, the results are almost always easy to predict.
- **Collision detection:** this lets you prevent cuts where the tool taper or collet holder would hit the workpiece, because you miscalculated the reach of the tool. This is useful chiefly if you have a machine with an automatic tool changer; otherwise, adjusting collision detection settings after every tool change is about as problematic as configuring the toolpath correctly to begin with.
- **Support for workpiece approach methods:** center-cutting tools can work as drills, but aren't particularly competent in this role; plunging them directly into tough materials may reduce their life expectancy quite a bit, especially if the cutter has a flat tip. This can be solved by generating toolpaths where the tool enters at an angle - that said, for easily machinable plastics, it's pretty much a non-issue.
- **Cutting wizards, libraries of materials, etc:** while these seem cool, they are generally tailored at simple industrial applications or other beginner work. Configuring the parameters manually is almost always a safer bet.
- **Drawing, part editing, and analysis tools:** having these capabilities available in a CAM package is useful only if you don't have them in your CAD application - but that, again, shouldn't be the case here.

Of course, feel free to ping me if you stumbled upon any other cool-sounding feature, and need help figuring out what it's worth.

### 3.2.1.8. All right - so which CAM package is good for me?

---

Well, it's complicated. For starters, your CNC machine may come with a basic CAM application, and that package may turn out to be good enough for starter jobs. For example, Modela Player 4 - the application that comes with Roland mills - is pretty decent. That said, in the long haul, you will probably want to upgrade to something more featured and flexible.

Somewhat lamentably, there aren't that many free-of-charge applications that would be easy to use yet powerful enough; [FreeMill](#) is one potential choice, but it only supports a single, rudimentary machining strategy - and really, isn't worth much.

This leaves you with commercial tools. If you are on a tight budget, you may want to check out [MeshCAM](#) (\$200), which is actually a very competent package with good technical support. Other alternatives within that price range include [Cut 3D](#), or the hobby license for [DeskProto](#). I have tried both DeskProto and MeshCAM, and both of them

are pretty good - so these would be my top picks.

Other than that, there are quite a few other reputable choices with prices hovering around \$1,000 and more. The pricing in the "pro" segment, especially for more featured packages with 4-axis machining, is still rooted in the era of strictly-industrial CNC - but deep discounts are available to students, and sometimes, to those who buy the software along with the mill itself. Notable applications in this category include [VisualMILL / RhinoCAM](#), [Alibre CAM](#), [Mayka](#), [madCAM](#), and quite a few more. If money is no object, or if you can get a good discount, you may want to dig deeper. If you need advice, I've seen VisualMILL in action, and it looked pretty good.

**Tip:** many of the commercial packages have demo versions. These demos are either limited to a 30-day trial, or lack the ability to write G-code, but are otherwise fully-featured - so be sure to check them out before you buy.

### 3.2.2. First minutes with CAM

CAM programs can be counterintuitive; you will probably need to read the manual carefully to even understand how to create a toolpath, or write the resulting NC file. These applications are also often finicky, unforgiving, and buggy - so it's important to take it slow, and check everything twice.

Before loading your 3D models, you need to configure the software for your particular machine, either by selecting the appropriate postprocessor, or - in the worst case - by writing your own based on an existing config file for a similar mill (not as scary as it sounds, as long as your machine came with a code reference manual). You should also go through all the configuration options, and make sure that units are set to millimeters, that the tolerance is 1 μm or better, that arc interpolation is enabled if supported by your hardware, and that the G-code coordinate system is set to G54 (the first, default user-configurable coordinate system, and the only one you will be using in hobby work).

With this out of the way, you should load the geometry to be cut. Let's use the heart-shaped mold created earlier on; if you need to export it from your CAD application to a mesh-based format, such as DXF or STL, be sure to configure meshing tolerance to 1 μm or better for that step, too, as this setting is often separate from the global value.

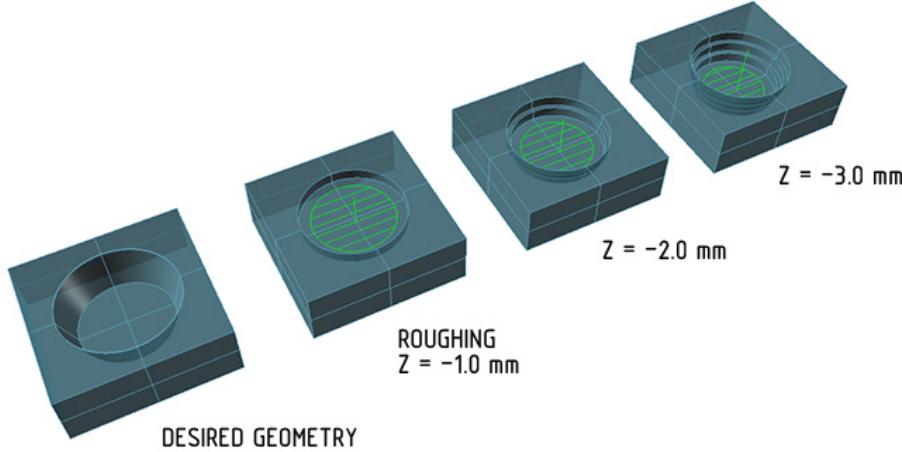
Once the model is loaded, you need to verify its position. For three-axis work, the mold cavity should be facing up, and the top surface should be aligned with the **Z=0** plane in the CAM application; when looking from the top, the bottom left corner of the mold needs to be at **X=0, Y=0**, too. If it wasn't loaded this way, you can move or rotate it as needed, but also try to figure out what went wrong: is your original CAD model oriented correctly? If that's not the reason, is there a CAM-level option you should have toggled at import time?

Well, all right. Once the model is in the right spot, it's time to create some toolpaths next.

### 3.2.3. Roughing toolpaths

To maintain sanity, it helps to split the cutting process into several phases. Each of these phases accomplishes a different task, and may use a different tool, feed speed, machining strategy, stepover distance, and so forth. The first phase of almost every project is known *roughing*; its goal is to remove the bulk of the material as efficiently as possible. Of course, astute readers may ask why this isn't also the final step - and the answer is that, quite simply, heavy chip loads and rapid tool speeds offer limited accuracy; it's better to leave a small margin of uncut material, and refinish the surface in a separate process later on.

The usual roughing strategy is a hybrid approach where the model is cut in Z layers, and on every layer, the application first uses a pattern of horizontal movements to clear the pockets, and then performs a waterline-type cleanup pass. Once the first level is done, the toolpath advances to the next one:



In some applications, roughing may not include that cleanup pass, in which case, you should probably configure a separate waterline step right after roughing. This is to ensure that the margin of remaining material is reasonably uniform, so that chip loads don't fluctuate wildly later on.

**Tip:** good CAM applications let you use pocket-shaped [offset cuts](#), rather than linear movements; this completely eliminates the need for the waterline cleanup pass. If that mode is offered by your software, take advantage of it - it will speed up the process and also let you maintain a consistent direction of cuts.

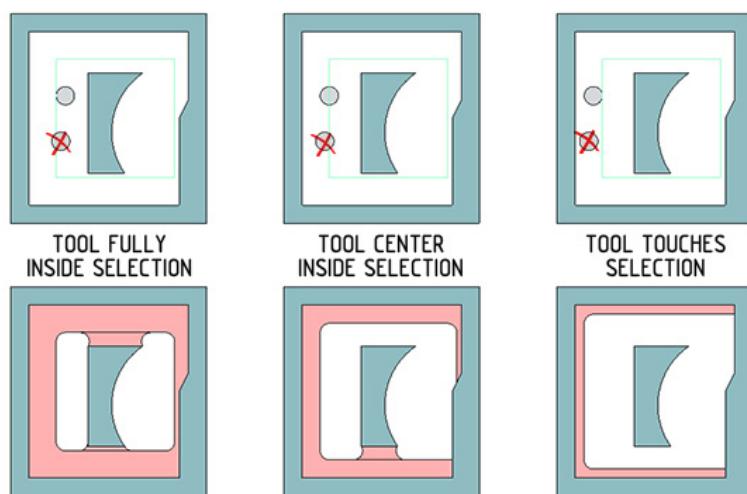
In any case, for the materials used in this guide, you should configure the roughing process the following way:

- **Tool size:** use a 3 mm or 1/8" flat tip cutter, but lie to the application and give it a diameter about 15% higher (say, 3.5 mm). This will cause the toolpath to be slightly farther away from the workpiece than actually makes sense for the tool - thus creating a margin for roughing in the X-Y plane.
- **Tool length offset:** tell the program to pretend that the tool is about 5% of tool diameter longer than set on the machine (effectively moving the toolpath up). This introduces a similar margin in the Z axis. (If not supported by the application, you can make this adjustment simply by setting Z origin a bit higher than measured.)
- **Roughing margin:** set to zero. Using tool size and length offset is usually more intuitive, especially for toolpaths where you want to have an offset only in a specific direction.
- **Stepover:** something between 30% and 50% of tool diameter should do. You can ramp it up even higher, but there's no real need to do so in this case.
- **Z level spacing:** set to 1 mm. Especially if the value is high, it should at least vaguely correspond to the spacing of key features of the mold in the Z axis.
- **Tolerance / sampling resolution:** for roughing, can be safely set to around 0.01 mm or so, to reduce processing time.
- **Machining speed:** use maximum spindle RPM; feed per tooth should be about 0.08 mm (i.e., 2,000 mm/min with four flutes at 6,000 RPM).
- **Cut direction:** when using offset cuts, use conventional cutting with the tool first engaging at the center of each pocket. When using linear cuts, set to bidirectional (*meander*) mode. No special approach strategy required.
- **Machining region:** set so that the cutter stays completely within the envelope of our mold, and does not attempt to trace around it. Note that depending on your CAM application, selection regions may work in one of three ways:
  - **Tool fully inside:** the cutter must stay completely within the bounds of the selection. Nothing that lies outside the region will be machined. Protrusions

that lie within the selected area, but are closer than `tool_diameter` to the boundary, will not be fully profiled, as the tool won't be able to pass between them and the limiting curve.

- **Tool center inside:** the center point of the end mill is required to stay inside the selection, but the tool itself can stick out slightly. Features up to `tool_diameter / 2` away from the selection can be machined by the tool if necessary. Protrusions within the selection must be at least that far from the boundary to be fully profiled.
- **Tool touches selection:** the tool is free to roam anywhere as long as any point of its geometry stays within the selected region. In this mode, features within `tool_diameter` from the boundary may be machined. Everything within the selection is cut.

It doesn't really matter which selection philosophy the application happens to use, as long as you are aware of its behavior, and can plan accordingly. If you are still not sure what's going on, check out this image; teal denotes the desired geometry, the selection rectangle is cyan, and the remaining unwanted material is red:



Well, that's pretty much it. Once the roughing operation is set up, simply tell the program to calculate the toolpath, and examine the result closely. Make sure that it actually makes sense: is the tool staying within our virtual workpiece? Do the toolpaths look anything like what we discussed before? If the program can give you estimated machining time, is it less than 10 minutes or so?

If everything checks out, congratulations; now, let's finish this thing.

### 3.2.4. Finishing toolpaths

Finishing toolpaths typically refine the geometry in several consecutive steps:

1. The vertical (Z axis) margin on flat surfaces is removed using a projection toolpath that has a zero tool height offset, but preserves the original tool diameter offset from the roughing process.
2. The X-Y plane margin around vertical features is removed using properly spaced waterline cuts, configured with the true diameter of the tool.
3. If there are any sloped surfaces, they are locally refined using a tightly spaced projection toolpath or 3D draping, possibly with a ball end tool (this is not applicable to our model).
4. If there are any holes or tight pockets that couldn't be faithfully reproduced with the current cutter, they are selectively refinished with a smaller one.

To take care of the first item on this list, you'd normally want to configure an offset-type, projection finishing toolpath with the following parameters:

- **Tool size:** about 2-5% higher than the value used for roughing (3.6 mm or so).
- **Tool length offset and margin:** both set to zero.
- **Stepover:** not critical; 1 mm is a safe bet.
- **Tolerance / sampling resolution:** 0.5 µm or so.
- **Machining speed:** maximum RPM; feed speed should be about 0.05 mm per tooth (i.e., 1,500 mm/min with four flutes at 6,000 RPM).
- **Cut direction:** conventional if using offset cuts. Otherwise, for parallel toolpaths, use bidirectional cutting mode to save time.
- **Machining region:** restrict to the inside of the mold cavity. Make sure that the generated toolpath doesn't climb the sides of the cavity, as this would be a no-op.

This step cleans up the flats; now, we need to take care of the walls. This is done with a "pure" waterline toolpath that traces around vertical shapes, and does nothing more. The parameters for this step should be:

- **Tool size:** enter the actual, measured diameter of the tool. It's good hygiene to include TIR, if known.
- **Z distance:** set to 0.5 mm. In general, this must coincide with the spacing of vertical features of the mold, and shouldn't be too large (to avoid high chip loads). Any features that are not aligned with this parameter may need to be refinished separately.
- **Machining region:** restrict to the inside of the pocket. This time, make sure that the outer walls of the pocket will be machined.
- **Starting height:** 0 mm. Always double-check this parameter for waterline toolpaths, as rouding errors may sometimes prompt the CAM application to default to something dubious.

Everything else should be the same as for the previous step.

Well, that's almost it. After this toolpath, the heart should be looking pretty much the way we designed it - with just one minor blemish: center kink in the middle of the heart will have a slight fillet on the top, due to our relatively large 3 mm tool not being able to fully squeeze into that tight spot. To fix this problem, we can perform a selective waterline cut with a 1 mm end mill. Most of the parameters don't change from step #2, except:

- **Tool size:** enter the actual, TIR-compensated diameter of the new tool.
- **Z distance:** use 0.25 mm.
- **Speed:** maximum RPM, feed of around 0.035 mm per tooth should be fine (that's around 450 mm/min with two flutes at 6,000 RPM).
- **Machining region:** make a rectangular or circular selection extending a few millimeters around the problematic kink.

And... well, that's it! Generate the toolpaths and pat yourself on the back.

**For future reference:** here's the list of maximum recommended cutting parameters for the tools you should have bought. These values are applicable only to several classes easily machinable prototyping materials that we're going to talk about soon - when working in more demanding stock, you will need to slow down:

- **0.4 mm:** feed 0.01 mm/tooth, 0.125 mm cut depth, 0.125 mm stepover,
- **1 mm:** feed 0.035 mm/tooth, 0.25 mm cut depth, 0.4 mm stepover,
- **3 mm:** feed 0.1 mm/tooth, 1 mm cut depth, 2 mm stepover,

- **6 mm:** feed 0.2 mm/tooth, 3 mm cut depth, 5 mm stepover.

*Speeds reduced by 20-40% are recommended for finishing operations, especially with 3 mm and 6 mm tools.*

### 3.2.5. Final sanity check

Measure twice, cut once. Before sending any data to the machine, it's always good to recheck your work; 5 minutes of that may save you several hours of troubleshooting in the physical realm. Here are some questions to ask yourself:

- **Have you entered all the values correctly?** Confirm tool diameters, stepover, Z distance, and feed speeds. A missing zero or a misplaced decimal point can cost you \$20 for a broken tool.
- **Do the toolpaths look OK?** Anything that seems weird, suboptimal, or that exits the expected machining envelope? Any places where the toolpath seems to go straight through the geometry? Run a simulation if possible, and inspect the model from all sides.
- **Have you covered everything that needs machining?** Are there any spots where toolpaths do not reach, even though you wanted them to? For selective refinishing, have you selected all the regions you wanted to clean up? It's embarrassing to miss a spot.
- **Is there any risk of collisions?** Are the toolpaths suitable, given the reach of the tools you are going to use? Are all the traverse movements happening well above the workpiece? Several CAM programs make it easy to misconfigure this, and plunge through an [essential part of your work](#).

Try to use that checklist until you are reasonably comfortable with the process; it's not that mistakes are common, but they can strike at inopportune times.

If everything checks out, you need to get the toolpaths ready for cutting. In some programs (e.g., Modela Player), the output is done directly from the application - in which case, just sit tight. In many other apps, the data is written to a text file, and that file is then sent to the machine using a separate CNC utility; if so, export ("post") the toolpaths now. Of course, remember to save the toolpath for the 1 mm tool in a separate file!

## 3.3. Please, let's cut stuff already!





Okay, okay - but first, you need a suitable workpiece. We will cover more permanent options shortly, but for your initial tests, I strongly recommend getting something called *machinable wax*. It is a hard, rigid, wax-like substance, technically a blend of low-density polyethylene (LDPE) and paraffin. It is not the cheapest or most durable stock, but has two important properties: it machines quickly to a very good finish with excellent dimensional accuracy; and more importantly, because it's much softer and more fragile than common plastics, even fairly major mistakes won't immediately result in a broken tool.

In the States, the material is available cheaply from the folks at [MachinableWax.com](http://MachinableWax.com). I would suggest ordering their set #17-424215 (featured on [this page](#)). It's a box of 12 pieces, 42 x 42 x 15 mm, selling for \$21. Don't overdo it - this really isn't the best or the most cost-efficient material for real work, so get just a single box or two.

The only other thing you need in advance is a way to secure the wax to the milling table. Your mill may have come with some sort of a clamping system, or at least slots or screw holes that let you rig something together after a quick trip to the hardware store. Alternatively, you may want to consider [MWHS01](#) from High Tech Systems (\$100), which is a very versatile and lightweight system of clamps that should work for almost any medium or large CNC mill. That said, when working with easily machinable materials and using small tools, you can also get away with strong, thin double-sided tape, such as [3M 444](#) or [Tesa 4965](#). Simply place four short strips in the corners of the workpiece, and press it down; then lift one of the corners to detach.

### 3.3.1. Setting up a job

Assuming you have all the necessary supplies, we can mount the workpiece at this point. Wipe clean the table and the stock material, apply the double-sided tape as described earlier, and attach the workpiece to the table, making sure that it's entirely within the working area of the mill. Depending on the design of the machine and the length of the tool, you may need to put something underneath the workpiece to allow the cutter to reach it, too.

Next, confirm that the workpiece is actually attached securely, and install your 3 mm tool in the collet. Tighten the tool holder, quickly confirm the TIR, and make sure that there is some clearance between the tool and the top of the workpiece when the spindle is fully retracted. Assuming everything is fine, you can now turn on the mill and let it initialize.

**Caution:** I am assuming that, as requested earlier on, you have familiarized yourself with the operating manual of your mill, and with the safety tips information included in [section 7](#) of this guide. As with any power tool, you can get hurt if you are careless.

For G-code mills, the next step is to set the coordinate system to G54, then move the tool to align it with the bottom left corner of the workpiece (looking from the top), plus about 5 mm. With the tool at this X-Y location, tell the machine to set its X and Y origins; this is done in a hardware-specific fashion.

Setting the Z origin is the last hurdle to deal with. Find a convenient location on the workpiece, in a region that won't be removed in the cutting process. If the machine has a tool sensor, simply place it underneath the tool, and follow the manual to perform the measurement. Without a sensor, you can use the trick we talked about earlier on: place a thin strip of paper or foil on top of the workpiece, and wiggle it back and forth while lowering the tool; use increments of 0.05 mm when you are close, and stop the moment the tool pinches the foil. Set the Z origin in that

location; oh, and try not to pinch your hand instead!

Done? Well, in that case, it's time to put on safety glasses, perhaps put several dabs of cutting oil in the region to be cut, and then hit "send" and output the 3 mm roughing and finishing toolpaths to the machine. With any luck, several minutes later, the machining process should be wrapped up. Vacuum off a bit and inspect the result; is the shape in line with what we're expecting? If yes, let's fix the kink: install the 1 mm tool, check TIR, redo the tool height measurement (**very important**), and hit "send" to output the remaining toolpath on your list.

In case you are wondering, there are very few things that can go wrong with the cutting process itself, and all of them should be fairly self-explanatory. For example, if the machine starts cutting at an unexpected location, or too high / too low in relation to the workpiece, you probably messed up setting the origins, selecting the right coordinate system, or had the CAM model positioned incorrectly. Similarly, if the tool breaks while cutting in the expected location, this is probably due to grossly incorrect RPM, feed speed, or cut-in depth; or bad ordering of toolpaths. But chances are, you won't run into any issues just yet, so sit back and relax!

*Note: as hinted earlier, for more complex jobs, it's useful to keep the roughing end mill and the finishing cutter separate, even if both processes call for the same diameter of the tool. This way, you don't have to worry about the loss of micron-scale accuracy as the roughing end mill begins to wear - and you will be able to use both tools for a lot longer.*

### 3.3.2. Inspection and troubleshooting

As soon as you're done, detach the mold and vacuum in thoroughly. Machinable wax sometimes requires a gentle but firm cleaning with a brush and a touch of compressed air to get rid of the somewhat sticky chips - so do that, and then have a closer look. Is the surface silk smooth? Do you see any pronounced tool marks or gouges on vertical surfaces? What about horizontal ones? Do the dimensions check out when you use a caliper? (In more complex projects, pin gage sets are useful for measuring the diameter of small holes, too.)

Chances are, you're in good shape - but if anything is even slightly wrong, now would be the right time to track the problem down; don't wait until you're working on more time-consuming projects, or in less forgiving materials. Here are several causes of the problems you are most likely to see:

- **Tool ploughing through an essential part of the geometry:** this may be due to poorly configured toolpaths; failure to adjust Z origin after tool change; axis stepper motor overload in open-loop machines; or sending a cutting job while the tool is positioned below the top of the workpiece (not all CAM applications prefix the generated toolpath with an instruction to retract the cutter). Double-check that the workpiece is still properly attached, too.
- **Significant dimensional inaccuracy:** this is usually caused by incorrectly specified tool diameter, excessive TIR, misaligned spindle axis, incorrectly secured workpiece, or poor mill calibration. The last problem will be directly proportional to the size of the machined feature, and if so, can be measured and compensated for in software.
- **Substantial tool marks or gouges:** these defects are typically due to high TIR, excessive speeds, or high chip loads during finishing (e.g., due to incorrect choice of toolpaths). Sometimes, lowering RPM is all you need.
- **Very poor overall appearance of the mold:** this may be due to unsuitable stock material (e.g., certain types of soft, malleable thermoplastics); using a bent or chipped tool; using a drill instead of an end mill; or having the workpiece partly detach from the table in the middle of the cutting job.

Still, in all likelihood, the result will be fine. If that's the case, let's wrap up by reviewing some common-sense rules that come handy in mold design, just so that you are well-equipped to work on your next CNC project - and then, let's cast a mold.

### 3.4. Things to avoid in mold design

There are relatively few real constraints on what you can cut on your milling machine, but there are several simple

design strategies you should follow to keep the cutting process simple and quick, and make accurate, durable, and dimensionally accurate molds. Here's the gist of it:

- To speed up toolpath calculations, always choose straight lines over arcs when you can. For example, you may want to opt for simple, manually drawn shapes for mold cavities, or use chamfering instead of filleting to provide reliefs in places where aesthetics do not matter. This keeps the resulting 3D mesh a lot simpler, and can dramatically improve the speed of CAM work.
- To simplify the cutting process, it's good to maintain a reasonably consistent margin between each of the parts inside the mold, and between parts and the walls of the mold cavity. Use a distance that is slightly higher than the diameter of the tool entered for the roughing process.
- To help with waterline cutting, it's useful to align most of the vertical features present in the mold to 0.5 mm or so. This minimizes the need for selective remachining of individual parts.
- When dealing with parts of vastly different part heights, it may be desirable to create several separate mold cavities, instead of trying to cram everything into a single one. Using a shallow cavity for all the small parts will often save you time and materials.
- It's preferable to avoid holes and slots that have a size exactly matching the diameter of the tool that will be used to cut them; due to rounding errors, such situations can confuse your CAM software, and make it hard to generate a valid toolpath. Allowing at least 1% wiggle room is a good idea.
- Any small, deep holes that reach the very bottom of the master pattern should have their depth reduced by about 0.1 mm. This eliminates the risk that the resulting protrusion of the flexible negative mold will rub against the top cover used in the casting process, and deflect. The resulting 0.1 mm film is very easy to remove from the final part.
- You need to allow some headroom between the top of the mold cavity, and the top of the tallest part laid out on the inside; this forms the bottom surface of the resulting negative mold. Allowance of around 2-3 mm is usually OK.
- Deep molds (over 10-15 mm or so) benefit from draft angles on their outer walls (a slight slant, created using the offset and loft operators). Introducing several 5 mm deep pull tabs on the edges of the mold may be helpful, too.
- Large-volume parts should use molds that are scaled up accordingly, and have more generous margins and headroom, to support the weight of the resin and the mold itself.

Well, that's it. It may sound overwhelming, but once you catch a whiff of it, even remarkably complex multi-part molds are fairly easy to crank out. We'll cover this and several other advanced design topics later on - but for now, let's have some fun with polymers!

## 4. Resin casting and you

---

As discussed in [section 1.3](#) of this guide, [resin casting](#) is a pretty amazing, simple, and user-friendly process that comes equally handy in CNC prototyping, 3D printing, and in manual DIY work; all its sophisticated uses aside, you will end up using it to replace broken parts in appliances or toys, make unique gifts, or even encapsulate backyard flowers or bugs (don't deny it).

Alas, the online market for moldmaking and casting supplies is dominated by several companies that cater chiefly to artistic users, and sell expensive products with poor mechanical properties and little utility in high-precision engineering work. In that spirit, even if you are familiar with artistic resin casting using epoxies, polyester resins, or polyurethanes from [Alumilite](#), [Smooth-On](#), and similar sources, you will be probably surprised by how much better your results can get.

### 4.1. Raw materials

---

#### 4.1.1. Machinable stock

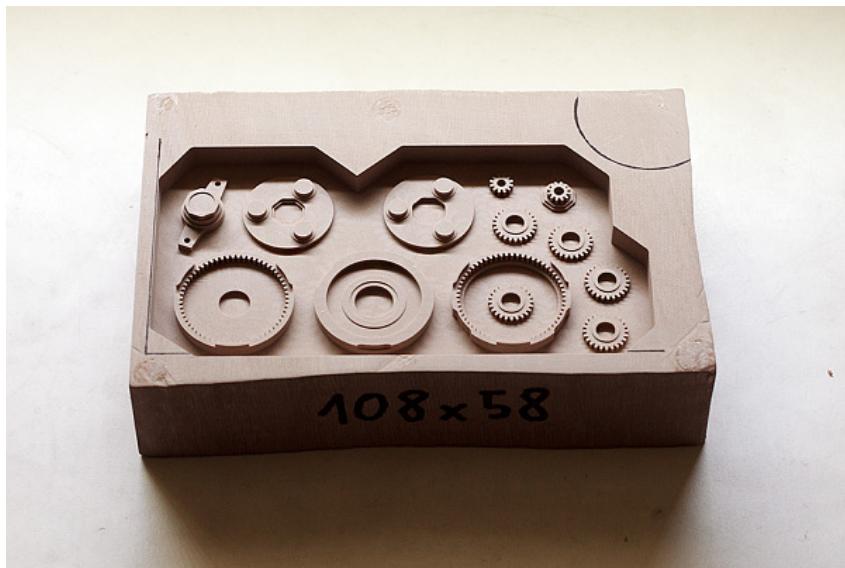
---

Well, before we dive into the world of casting resins, we should briefly revisit the choice of materials you can use to make master patterns. This part of this chapter is specific to projects that rely on CNC milling, so if you are interested in replicating manually crafted or 3D printed parts, you may want to skip ahead a page or two ([this site](#) provides a good overview of how to build patterns by hand - and if you have any other questions, feel free to drop me a mail or stop by [/r/resincasting](#)).

Still here? All right! Of course, milling machines are not particularly fussy, and will cut almost anything that is softer than tungsten carbide, but rigid enough to stay in one place; still, some materials are more predictable than others, and produce better results. Prime choices include rigid engineering plastics such as polyurethane, epoxy, polyester, ABS, polyamide (Nylon), or acetal (Delrin); many varieties of hard woods; aluminum, brass, and other soft metals and alloys; and more exotic choices such as printed circuit boards, hard waxes, plaster, etc.

Conversely, common materials that machine with greater difficulty or offer sub-par surface finish include stringy, low-melt thermoplastics (PET, polycarbonate, some grades of polyethylene); rubbers and other stretchy or squishy polymers (including some grades of PVC and most polystyrene foams); plywood and particle boards (including MDF); and exceptionally hard or highly abrasive stuff, such as steel, stone, or glass. Of course, many of these materials can be still cast or formed using CNC-machined molds and dies.

Poor choice of working materials is one of the most common mistakes made by hobbyist machinists; quite a few people stick to workpieces that offer poor accuracy, get damaged easily, gum up the tool, or simply cost way too much. For moldmaking purposes, your best bet is one of the little-known materials: an extremely accurate, low cost piece of plastic known as a *medium-density modelling board*, originally devised for the automotive industry. It's essentially a mix of medium strength polyurethane, and a combination of soft fillers such as calcium carbonate and aluminum hydroxide. There are many types of machinable boards, but the one we are interested in has a density of about 0.70-0.78 g/cm<sup>3</sup>, and vaguely resembles wood:



Prototyping boards of this particular variety include [Huntsman RenShape 460](#) (or slightly less dense BM 5460), [Axson ProLab 65](#), [Sika SikaBlock M700](#), [BCC MB2001](#), [Necuron Necumer 651](#), and several more. The material is typically sold in bulk, in sheets of about 50 x 150 cm, 25 mm thick. This may sound like a lot, but I recommend buying a full board, rather than grossly overpriced cut-to-size bits. The material lasts me for about a year, and costs about \$12 per liter (roughly \$250 for the whole thing). It's much less than what you'd pay for a similar slab of HDPE or acrylic - and it machines easier, too. (Planks of dense hard wood, when glued together and planed, may be a cheaper alternative for uncomplicated parts.)

Buying prototyping boards is actually pretty easy. If you are in the US, you can simply go to [Freeman Supply](#), and order RenShape 460 online (search for item #075229). In other places, simply look at the manufacturers' websites and find local distributors, then send out several e-mails or make some calls (online ordering isn't common in the industrial world). Be aware that prices may vary significantly, so shop around.

Alongside with the board, you may want to order a matching board repair putty; it's a fast-curing, polyester compound that can be used to fix minor damage to your molds, or even completely fill a previously created cavity to

reuse a particular workpiece for a new project. If you are ordering online with Freeman, go with their [Quik-Fil](#); otherwise, ask the distributor for a matching product - they will be able to advise.

What else? Oh, about the only minor drawback of the medium-density boards is that they have a very fine but perceptible grain, as shown in this magnified image:



This grain normally has no appreciable effect on dimensional accuracy, but imparts a satin finish that will transfer to any transparent, water-clear parts. Of course, you can create high-gloss molds by coating the pattern with paste wax or a similar sealer (carnauba wax is particularly good); or you can always simply polish the final part - but both these options affect dimensional accuracy, and can be annoying when working on complex molds. The alternative is to use a more expensive material known as a *tooling board* - made out of solid, dense polyurethane, with no perceptible grain. Boards such as RenShape 5169 or BM 5272 cost up to 50% more, need to be machined 20-40% slower, and cause some wear to the tool - but they scratch this particular itch.

**Note:** RenShape 460 is relatively easy to cut with a hand saw; in fact, it's comparable to soft woods. That said, the extra labor may be annoying in the long haul, so it makes sense to have a decent jigsaw nearby. You can get one for around \$35; blades designed for hard woods will cut the material very quickly and last for years.

#### 4.1.2. Silicone rubbers





In order to replicate the parts laid out inside your pattern cavity, you need a flexible and durable substance to take an impression of the desired shape, and use it as a mold for the final product of your work.

There are several types of castable rubbers that could be useful for this purpose, but silicones are hard to beat. There are quite a few formulations that combine ease of use, excellent mechanical properties, perfect dimensional accuracy, no odor, no toxicity, and temperature resistance up to 300° C. On top of that, silicones come with an inherently non-stick surface, which helps greatly in casting work.

Almost all the silicone formulations you can find on the market come as a viscous goo consisting of long, linear, partly polymerized chains of siloxanes; that nominally non-reactive soup is then combined with a suitable cross-linker and a catalyst. The reaction between these components quickly turns the goo into a very bouncy solid; this can be initiated in several ways:

- **Room temperature vulcanizing, one component rubber (RTV-1):** these substances undergo hydrolysis when exposed to atmospheric humidity. This reaction creates unstable molecules that promptly bind to each other, and release acetic acid or a simple alcohol as a byproduct (hence the trademark smell of the silicone sealants from a hardware store). This reaction, known as *condensation polymerization*, has the unfortunate effect of subtly affecting dimensional accuracy of the part - simply due to the release of volatile molecules. That said, the main reason why this process isn't popular for moldmaking is more prosaic: it's just that thicker layers take forever to set, owing to their limited exposure to air.
- **High-temperature vulcanizing silicones (HTV):** here, condensation polymerization occurs in presence of free radicals, which are liberated from a heated organic peroxide. The suitable curing temperature hovers around 100-150° C. This method is sweet and simple, but also has a fatal side effect: the coefficient of thermal expansion for silicones is pretty high, around 0.025%/°C - and so, after cooling down, the dimensions will be off by several percent. This makes HTV rubbers completely useless for precision work. (It is possible to use radiation or UV light instead of heat, but that's not very practical in most cases.)
- **Room temperature, two-component rubber (RTV-2):**
  - **Condensation cure:** these rubbers polymerize at room temperature in presence of an organotin compound (typically dibutyltin dilaurate), which is mixed into the system shortly before pouring it all into a mold. The catalyst itself is fairly harmful, has a characteristic smell, and will leach out of the rubber in small amounts. Tin-cured silicones are popular in hobby work, but I don't recommend them; organotin aside, they exhibit measurable shrinkage (around 0.5%), have inferior mechanical properties, and are prone to cure reversal after 1-2 years or so.
  - **Addition cure:** these silicones are catalyzed with a platinum complex, and polymerize with no byproducts whatsoever. The materials are safe and odor-free, and the finished product has practically no shrinkage, exhibits superb mechanical characteristics, and can be stored indefinitely. Platinum cure

rubbers are commonly used in medical and food handling applications, and are the prime choice for flexible molds.

In other words, you almost certainly want to stick to RTV-2 platinum-catalyzed silicones, unless you are working on life-sized castings (at that point, the cost of silicone can become prohibitive).

Before discussing specific products, let's have a quick look at the notable characteristics that will come up in product datasheets for these rubbers, and review their significance to our work:

- **Stiffness:** the rubber we are going to use needs to be flexible enough to allow easy demolding of finished parts, but must be sufficiently rigid to maintain dimensional accuracy when laid on a flat surface, filled with liquid resin, and clamped or weighed down until the plastic is fully cured.

There is no single, consistently advertised parameter that would give you a good idea of how flexible a particular rubber is, but this can be inferred in two relatively simple ways:

- **Indentation hardness:** this parameter is measured using an ad hoc scale known as "Shore A", named so after its inventor. The test involves pressing a flat-tipped 0.8 mm needle into a sample with a modest force (800 gf). Deflection of 2.5 mm or more corresponds to 0 on the scale, while no deflection whatsoever is denoted with the value of 100.

Indentation hardness is a relatively poor proxy for rigidity, but is the primary key by which silicones are sorted in any catalog. In general, silicones under 10 Shore A are jello-like, and are used chiefly for special effects, soft bicycle seats, and - why not - sex toys. Compositions around 20-40A are fairly stretchy and squishy, similar to a rubber band. Silicones around 60-80A are still flexible, but begin to resemble a pencil eraser or a tire - you can flex them, but you won't stretch them easily.

**What to buy:** rubbers starting from 20 Shore A are marketed as suitable for moldmaking, but I strongly recommend using around 60A for small, dimensionally accurate molds; rubbers under 40 Shore A are not advisable, except for large-scale work or for certain manual moldmaking techniques.

- **Tensile strength vs elongation at break:** if you need a more accurate picture of flexibility, looking at these values is a better approach - especially when comparing silicones to other types of rubber-like elastomers that we'll talk about soon.

In essence, tensile strength tells you the *stress* (force divided by the area of cross-section) that causes the material to either snap, or to develop a localized defect known as necking. Elongation at break, on the other hand, describes the extent to which a standardized specimen can be stretched before breaking apart. If you divide tensile strength expressed as MPa (1 MPa = 145 psi) by elongation at break expressed as a ratio ( $1 + \text{elongation} / 100\%$ ), you will get a somewhat arbitrary but useful value that will be low for squishy rubbers, and high for rigid ones.

(There is also a proper way to measure stiffness - Young's modulus - but it's not very commonly seen in silicone datasheets.)

**What to buy:** unless you are working with very large or very deep molds, try to select a rubber where this calculated "stiffness coefficient" is at least 2 MPa. Ratios up to 3 MPa are OK for most uses; values around 4 MPa make sense only for relatively shallow molds where dimensional accuracy is paramount.

- **Tear strength:** removing cured parts from the mold inevitably involves some non-uniform pulling in the vicinity of sharp corners and other tight spots. In these cases, tensile strength does not accurately describe the resilience of the molding material; tear strength testing, which involves pulling

a sample apart near the edge, offers a much better insight. The test used in the States is usually [ASTM D624 die B](#), which involves a sample nicked with a razor; in Europe, a different method, similar to ASTM D624 die C, is more popular. There is no general mapping between these tests, but for moldmaking silicones specifically, the results given by the European approach (BS903 part A3) tend to be 30% lower or so.

**What to buy:** tear strength of at least around 15-20 kN/m, as per ASTM method, is highly desirable (the value corresponds to force divided by the thickness of the sample; 1 kN/m = 5.7 ppi). Formulations up to 30 kN/m are available on the market, and are certainly not a waste of money. It's best to stay away from products where the advertised tear strength less than 12 kN/m or so.

- **Mixed viscosity:** rheology of a liquid is a complex topic and remains very difficult to fully parametrize, but within a single family of products, formulations with lower dynamic viscosity are generally easier to mix, pour, and degas - and will be more inclined to conform to complex shapes without the aid of vacuum or pressure.

Silicone rubbers suitable for moldmaking start around 10,000 mPa\*s; this viscosity resembles a runny syrup, and is very easy to work with. Compositions that go over 35,000 mPa\*s or so are more prone to air entrapment when mixing the components, and when pouring them into a mold; the use of vacuum degassing is advisable for any complex work. Finally, once you cross 100,000 mPa\*s, almost any silicone becomes fairly painful to work with.

**What to buy:** all other things being roughly equal, go with the system with the lowest viscosity. Products below 50,000 mPa\*s are advisable, but not strictly a must.

- **Pot life and demold time:** RTV-2 formulations begin to polymerize the moment you mix the two components together; the point at which it the mix becomes so thick that it is no longer free-flowing or self-leveling is known as *pot life* or *gel time*. In the same vein, the point at which it is sufficiently cross-linked to be safe to remove from the master is referred to as *demold time*. Both of these values must be chosen so that the product can be worked with at a reasonable pace.

**What to buy:** pot life of around 30 minutes in the bare minimum for low viscosity compositions; for medium and high viscosity, 50-90 minutes is preferable, because every step of the casting process will take a bit longer. Demold time should be as short as possible; I'd stay clear of any formulations that require more than 24 hours to set.

**Note:** the speed of this and many other chemical reactions roughly doubles with every 10° C.  
If you are in a rush and don't care about some minor thermal expansion, using moderate heat is a great way to have your molds ready in 2-3 hours or so.

- **Shrinkage:** addition cure silicones used for moldmaking purposes should have no measurable shrinkage. Some manufacturers simply say "none", while others say "less than 0.1%" or so. Seeing a higher value in the datasheet is a warning sign; shrinkages between 0.2% and 0.6% are commonly seen in condensation cure silicones, and should be avoided in precision work.
- **Color:** moldmaking silicones are usually opaque beige in their virgin state, and can be pigmented easily. Systems in such a neutral color come handy if you wish to use the rubber not only for molds, but also to make functional parts (tires, etc). Sadly, many of the moldmaking compositions are artificially dyed blue, red, or green, to help novice users mix the components properly.

That aside, there are several silicone formulations that are nearly transparent. The product is never water clear, and will have a slight milky haze; it will also lose transparency if exposed to high temperatures. Because of this, it's not particularly suitable for decorative purposes, but can be indispensable for non-CNC replication work, where you often need to see the submerged master part, and carefully cut the mold to recover it. The major downside of transparent silicones is that they have somewhat different flow characteristics, and can be difficult to degas.

**What to buy:** I would not consider color to be the most important factor - if necessary, polyurethane

elastomers offer a good alternative for making functional parts. That said, if you have a choice, it certainly doesn't hurt to go with something that can be pigmented easily.

All right, that's it! Other parameters are either uninteresting, or are not advertised consistently. To help you with the selection process, my top recommendations for mechanical projects would be:

- **Best overall properties:** [Quantum Silicones](#) QM 262. This is a pale blue, opaque rubber with low viscosity (35,000 mPa\*s), hardness around 65A, 12 hour demold time, and high strength. [Hobby Silicone](#) is a reputable distributor, so give them a try.
- In Europe and some other markets, [Bluestar Silicones](#) RTV 3460 is a decent, but considerably more viscous, alternative.
- **Transparent / pigmentable:** [Silicones Inc](#) P-592 (formerly known as XP-592). Truly exceptional tear strength, good price, medium viscosity (50,000 mPa\*s), hardness in the vicinity of 60 Shore A, more challenging to degas. Cures in about 12 hours. You can order this product through [Innovative Polymers](#), along with some of the other items we'll cover in the next section.
- **High rigidity:** Quantum Silicones QM 270, also sold as [ACC Silicones QM270](#) in Europe. Neutral beige or turquoise rubber around 75 Shore A, excellent mechanical properties and manageable viscosity (50,000 mPa\*s). Remains easy to degas. Somewhat less suitable for deep molds, undercuts, and in other applications requiring a stretchy material - but its great for small, high-precision parts.
- **Super-low viscosity:** Quantum Silicones QM 237. Around 40 Shore A, very easy to mix and pour (10,000 mPa\*s), properties are still acceptable for typical molds. If you live in Europe, you may want to check out [ACC Silicones](#) MM242, which is pretty similar and comes in a transparent variety.

You should pick just one of these; if you're undecided, go with QM 262 or (X)P-592. If you need other options... well, Silicones Inc and Quantum are the most interesting US-based companies I know of. Other choices include [Polytek](#), [GT Products](#), [BJB](#), and [Smooth-On](#) - but in my opinion, their selection is much less impressive, and the options I have tried pale in comparison with the ones mentioned on the recommended list.

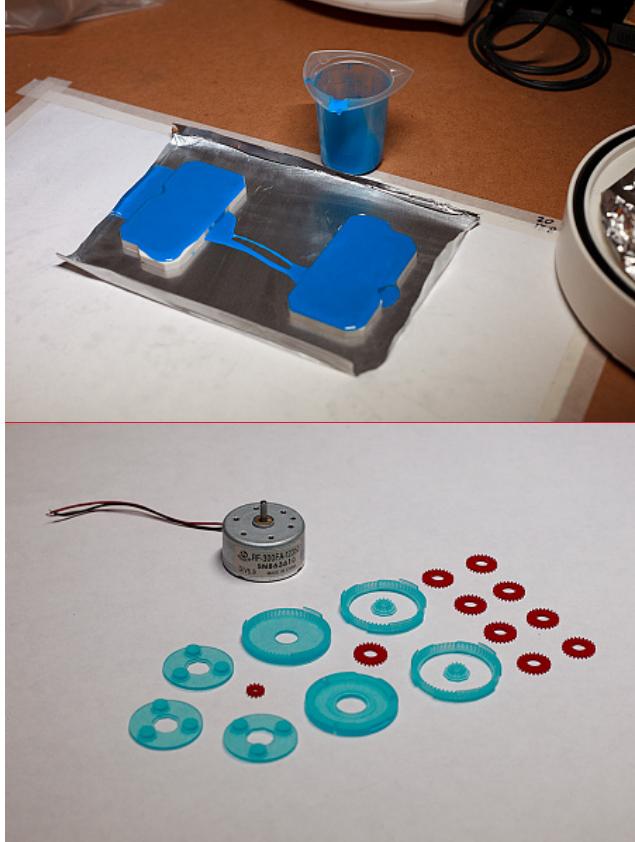
US market aside, globally, [Bluestar Silicones](#) (Rhodia) is fairly ubiquitous; you may also want to check out [ShinEtsu](#), [Wacker](#), [Zhermack](#), [Huntsman](#), [Axson](#), or [Dow Corning](#) - depending on where you are located, they all offer some interesting choices. But then, one reader living in Norway reported that placing an international order with Hobby Silicone for QM 262 actually turned out to be much cheaper than locally available alternatives (whoa).

As far as pricing goes, platinum cure silicones cost around \$35-\$45/liter when bought in one liter cans, or about \$30-\$35/liter in 4-5 liter pails; for example, (X)P-592 costs around \$120 for 4 l, while QM 270 and QM 262 in 5 l quantities fetch \$140 and \$170, respectively. Unreacted liquids should survive at least 2-3 years without significant deterioration, as long as you keep them in tightly closed containers, away from sunlight, moisture, and excess heat - so getting a full pail is not a bad idea.

*Note: resin manufacturers in the States use a somewhat confusing scheme for describing the size of their two-component kits: "1 gallon kit" usually means that you are getting about one gallon of whichever component is needed in greater quantity; and a matching amount of the other one. If the mix ratio is 10:1 (as is the case with most platinum cure silicones), the gain is minimal - but for resins mixed 1:1, you are actually buying two gallons or so.*

*To further confuse you, the same does not apply to sizes specified in pounds - "15 lbs" means that you are getting just enough to cast a 15 lbs blob of plastic or rubber. Be sure to account for these differences when shopping around: a lower price is not always a better deal.*

#### 4.1.3. Rigid polyurethanes



All right, let's talk about the materials you can employ to actually make final parts!

Polyurethanes are an incredibly interesting and versatile class of two-component, addition cure polymers. They use two principal reagents, mixed in comparable quantities: a non-volatile isocyanate and a complex alcohol (polyol). Some formulations trade some or all of the polyol for a polyetheramine, resulting in a material that is more properly called a polyurea. In any case, the two components are usually combined with variable amounts of chain modifiers, usually chemically similar to the primary polyol; and possibly surfactants, plasticizers, fillers, and so on. The whole thing is then catalyzed with a wide variety of organometallic compounds (bismuth, zinc, tin, zirconium, aluminum, or similar); with tertiary amines such as 1,4-diazabicyclo[2.2.2]octane (DABCO / TEDA); with diazoles such as 1,2-dimethylimidazole (DMI); or with something pretty close to that.

Modern polyurethane chemistry lets you manufacture everything from soft foams, to high-performance rubbers, to faithful, often superior imitations of many other rigid engineering plastics - all that without having to go bankrupt on injection molding equipment, and while using only fairly safe and predictable chemicals. They greatly outperform more familiar resins, such as epoxies or polyesters, and in hobbyist workshops, are much less dangerous to work with.

The only downside of high-performance polyurethane systems is that they generally require a basic vacuum rig - a small pump and a suitable container to remove any dissolved gases from the mix. Products that do not require degassing are readily available, but usually don't perform as well as their peers. (We'll talk about the required hardware later on, but it's pretty compact and doesn't cost a lot.)

Anyway - if you are aiming to make functional prototypes, it is probably prudent to start by stocking up on a polyurethane resin that lets you produce hard, rigid, and shock-resistant parts. Once more, let's have a look at some of the key things worth highlighting in a datasheet:

- **Stated purpose:** there is a lot of variety in the world of polyurethanes, and not all the properties of a material can accurately be captured with several cold, dry numbers alone. You should always pay attention to what the manufacturer is trying to tell you: if they are talking about simulating engineering plastics such as ABS, polyamide, or polyolefins - great. If they are instead paying undue attention to tooling fixtures, conceptual prototypes, scale models, etc, be wary. This may mean that the resin has disproportionately low strength in thin sections, is very brittle, or suffers from other malady. If in

doubt, simply talk to them and clarify.

For the same reason, tread carefully if the resin is designated strictly for meter-mix machines, vacuum assisted casting equipment, or something of that sort. It may mean that the material is very difficult to degas, has an annoyingly short pot life, or requires a heated mold. That label is not always a problem, but 75% of the time, it's there for a reason. As usual, when in doubt, ask. It's often also possible to get a sample and try it out yourself.

- **Hardness:** indentation hardness of rigid plastics is typically measured using a scale called "Shore D". This approach is very similar to the one used for rubbers, except that it relies on a sharper tip (0.1 mm) and a greater force (4,500 g). Indentation hardness doesn't necessarily map to rigidity, or any other useful property - but since it's always listed prominently, use it as a very rough selection criteria: plastics under 75 Shore D are unlikely to be strong enough for making thin-walled parts.
- **Flexural properties:** almost all the prototyping work you may end up doing will probably benefit from a plastic that withstands high bending forces without snapping in half; and that maintains as much rigidity as possible under such a load. These qualities are measured using two values: *flexural strength* and *flexural modulus*. The first one describes the force in relation to part dimensions at which a [standardized specimen](#) breaks; the other one tells you the ratio between the stress the material is under, and its degree of deformation. The higher it is, the more rigid the part will seem.

In comparison to these two, compressive and tensile properties usually matter a lot less in small-scale prototyping work.

**What to buy:** aim high. Look for flexural strength around 90-120 MPa, and for flexural modulus somewhere between 2.5 and 3.1 GPa. This is roughly comparable or better than tough materials such as ABS (Lego bricks), polyamide (plastic gears), polycarbonate (CDs and DVDs), or acrylic glass. For a comparison of common plastics, [this website](#) is a great starting point.

- **Impact resistance:** flexural strength of a polymer is important, but doesn't paint a complete picture. For example, polypropylene is a relatively flimsy plastic, with flexural strength less than 40% that of acrylic glass; yet, Tupperware made out of PP easily survives being dropped to the floor, while acrylic glass of the same thickness and shape would likely shatter right away.

The difference is that, especially in thin sections, polypropylene has a much better ability to flex and dissipate the energy of a localized shock; this is usually measured using a test known as [notched Izod impact strength](#), where a pendulum swings and breaks an upward-facing, notched sample in its path; the test estimates the energy lost by the pendulum, in proportion of the thickness of the specimen (minus the depth of the notch). Lo and behold - the strength of acrylic glass is barely 15 J/m ( $1 \text{ J/m} = 0.0187 \text{ ft-lbs/in}$ ), while various grades of polypropylene vary between 100 and 300 J/m.

Impact strength matters more often than you might expect. For example, tiny gears subjected to sudden torque variations when braking will be more prone to [failing catastrophically](#) if the material has poor resistance to such shocks. Most rigid polyurethanes have Izod impact strength around 30-50 J/m, resembling Nylon, polystyrene, and so forth: they will perform well, but if you step on a medium size, very thin-walled part, it will probably break. There are quite a few formulations closer to 100-300 J/m, resembling plastics such as ABS; if you ever stepped on a Lego brick, you probably know that it prevails, at the expense of your foot.

**What to buy:** your primary resins used for functional prototypes should have impact strength of at least 30 J/m; if the value is not provided by the manufacturer, it's probably lower than that. You typically don't need much higher figures, but if you have a choice, getting something closer to 100 J/m doesn't hurt.

- **Shrinkage:** this parameter is somewhat important in polyurethanes, because in some circumstances, it may significantly affect the dimensions of your parts. Alas, the standardized test is essentially meaningless. Polyurethane polymerization reaction has virtually no inherent shrinkage, but can be fairly exothermic; thermal expansion of the liquid, and the response of the flexible mold that surrounds it, is what may mess up the final result.

In general, coefficient of thermal expansion for rigid polyurethanes is about five times lower than that of silicone (perhaps 0.005%/°C); when using sensibly selected resins, the exotherm in thin parts will not exceed 5-10° C above room temperature, and the resulting shrinkage will be practically nil, no matter what the datasheet says. On the flip side, when dealing with parts that have thick cross-sections, the temperature may peak 50° C over ambient - and at that point, dimensional accuracy in the most easily deflected portion of the mold will suffer. For example, when casting a solid block of resin with dimensions of 10 x 10 x 30 mm, and using a sealed rubber mold and a resin with 10 minute pot life, the most vulnerable point in the mid-section of the mold may be off by as much as 0.3 mm (3%). The rest of the part will be probably spot on.

Luckily, it is within your power to change this. We'll talk about this in more detail later on, but in essence, you can:

- **Slow down the reaction.** The boring option is to switch to a system with a pot life of 30-50 minutes, and overnight cure - as these will usually have negligible exotherm even for bulkier castings. The more exciting option is to get rid of some of the catalyst present in the product, which usually involves adding a common and reasonably safe chemical to the system. It's simpler than it sounds.
- **Partly prepolymerize the resin.** Pre-mixing the full amount of isocyanate with about 10-20% of the intended amount of polyol, and setting this mixture aside for about an hour or two, will minimize the exotherm later on, likely reducing shrinkage by 70-90%. This has several other benefits, too, and the only side effect is increased viscosity.
- **Add an inert filler to act as a moderator.** Well-chosen additives allow you to reduce shrinkage by at least 30% without significantly compromising flexural properties of the system; going up to 80-90% is possible if you are willing to make some performance trade-offs.
- **Allow some extra resin to flow back into the mold as the system cools down.** Adding sprues or vents allows shrinkage to be offset without deforming the mold, and can result in dimensional accuracy improved by 80% or more.
- **Cast in several layers.** This eliminates shrinkage almost completely, but works well only with relatively simple shapes.

The approaches can be combined, often with synergistic results. It is also possible to let shrinkage happen, but direct it toward non-essential locations; this can be accomplished by intentionally weakening a section of the mold, for example by using thin walls. The bottom line is that you don't have to worry a lot - but you need to be aware of this behavior, and need to know how to reduce it when necessary.

**What to buy:** all in all, don't read too much into the datasheet, but be wary of resins with very high stated shrinkage - unless that shrinkage is measured by the manufacturer for a sample that is cured at an elevated temperature. For samples that spend the first few hours or days at room temperature, advertised shrinkage should be less than 0.3% or so. Beyond that, comparisons are somewhat pointless.

- **Mixed viscosity:** flow characteristics of polyurethanes are very different from silicones, possibly due to higher surface tension; on top of that, polyurethanes usually have much shorter pot life. Because of that, resins under 1,000 mPa\*s are preferred; products over 2,000 mPa\*s or so will be more challenging to degas and pour - and venturing past 4,000 mPa\*s should be done at own risk. The parameter is not critical, but it's good to avoid extremes.
- **Pot life and demold time:** many polyurethanes will have fairly short pot life, between 30 seconds and 4 minutes. That's usually not enough for manual casting, unless you are doing something very simple. Around 7-8 minutes is highly advisable for low viscosity resins, and closer to 10-15 minutes

makes sense for when dealing with resins over 1,500 mPa\*s or so.

As for demold time, it is usually preferable to keep it short; between 2 and 6 hours is ideal, as it lets you crank out multiple batches in a single day. That said, overnight cure (12-24 hours) is not the end of the world. Any slow-curing resin can be accelerated in a simple way, and we'll cover that a few sections down.

- **Presence of fillers:** some resins are pre-mixed with fillers to improve their flexural properties, abrasion resistance, or fire rating. In general, don't go there unless you are sure that you can't approximate this effect by manually adding the appropriate filler, and that you want to be permanently stuck with all the drawbacks (e.g., higher viscosity, lower impact strength, and the tendency for fillers to settle in storage).
- **Color:** avoid dyed or non-pigmentable resins. Transparent, translucent, neutral white, and slightly off-white resins (light beige, lightly straw-colored, etc) can be usually easily pigmented to your liking. Colors such as amber, caramel, or tan are more of a gamble; ask for a photo or a sample of the material to determine how bad it really is. Some of these colors are faint and easily overcome, and some are there to stay.
- **Compatibility with platinum-cure silicones:** there is a small number of transparent resins or other specialty formulations may be inhibited by the platinum catalyst present in silicone molds. Always scan the datasheet for any mentions of that risk. If it's not mentioned, you are good to go.

All right, ready for some recommendations? Here we go:

- **Best all-around resin:** [Innovative Polymers](#) IE-3075. Really, nothing else comes close. Translucent, high-strength plastic with a reasonable pot life (8 minutes) and quick demold time (2-3 hours for smaller parts). Around 83 Shore D, flex strength 120 MPa, modulus 2.9 GPa, decent impact resistance. Very low cost - around \$17 per liter. The system can be slowed down in a simple way if you need a longer cure time or lower exotherm.

A good fallback option in the US and several other markets is [BJB Enterprises](#) TC-854 A/B, which is about 15% less rigid and cures a tad faster, but is otherwise indistinguishable from IE-3075 and costs about the same. The other US choice is [Huntsman](#) RenCast 6591, but it has noticeably shorter pot life, is harder to pigment, and costs much more.

For those of you in Europe, RenCast 5146 (also sold as RenPIM 5219) is a very similar, low-cost resin. This one has a pot life of around 30 minutes, and requires overnight cure (unless you add some extra catalyst). There are some distributors for BJB products in the EU, too, so you may also have luck with TC-854 A/B.

- **Water clear:** Innovative Polymers OC-7086. Like other mercury-free compositions, it is extremely sensitive to moisture in thin layers. When making small castings in single-part molds, you may end up getting relatively low yields unless you preheat the mold, perform pressure casting, or add a small amount of a separately purchased catalyst (see [section 4.4.4](#)). The price of the resin is relatively steep, around \$45 per liter.
- **High-impact plastic:** Innovative Polymers TP-4052. This product is fairly similar to IE-3075 in most aspects. Flexural parameters are somewhat worse, but impact strength is almost three times higher. Price is in the vicinity of \$30 per liter. Note that as opposed to IE-3075, I have not found a way to slow this system down when low shrinkage in mass casting is desired.

At this point, unless you have a specific itch to scratch, it's perfectly OK to order just IE-3075 directly from IPI. Their "1 gallon kit" (actually around 6.5 l) sells for about \$110. "Quart kits" are also available if you need to try it out first.

If you want to shop around, I don't think it makes sense to look beyond Innovative Polymers - not if you're in the States; they have a remarkable selection of unique, user-friendly products designed specifically for manual casting, and great customer service. They take direct orders, and also have several local distributors. If you're skeptical, you can have a look at products from [Smooth-On](#), [Alumilite](#), BJB Enterprises, or Freeman - but you will not find anything that even comes close to that selection. If you are in Europe, checking out Huntsman and [Axson](#) is not a

bad plan.

As with silicones, polyurethane resins are pretty stable and have a fairly long shelf life - but they are fairly sensitive to sunlight, humidity, heat, oxygen, and moisture. It is a good practice to buy several 100-250 ml polypropylene or HDPE bottles ([example](#)) for intermediate storage of the amounts of resin you plan to use within a month or so - and keep the original containers sealed and blanketed with inert gas. The inert gas can be just a burst of "canned air" (difluoroethane or tetrafluoroethane), but if you want to save money in the long haul and have some room in your workshop, it makes more sense to invest around \$150 in a small nitrogen tank, a regulator, and a hose. This setup will last for months, and it's only about \$10 to refill at a nearby [Praxair](#) location or so.

In any case, if you store the resins properly, you can expect most of them to maintain their properties for at least 2-3 years; manufacturers usually give much more conservative guarantees, but take them with a grain of salt. Remember to agitate the containers if the components separate. Oh, some formulations may crystallize if kept below 10° C or so, but this process can be reversed easily. Prolonged storage in crystallized form is not advisable, as it may lead to the formation of insoluble dimers.

**Caution:** although most of the polyurethane formulations you will encounter are reasonably safe, there are some unfortunate exceptions. We'll talk about this a bit more in [chapter 7](#) - but for now, definitely watch out for:

- **Volatile isocyanates:** all isocyanates are dangerous if inhaled, but most of the prototyping formulations use prepolymers or monomers with negligible evaporation rates - so it would take serious effort to put yourself at risk (intentionally aerosolizing the resin would be one bad idea). In particular, MDI, HMDI / DMDI, and IPDI isocyanates are pretty safe. That said, some industrial resins use more volatile and more reactive compounds, such as toluene diisocyanate (TDI) or hexamethylene diisocyanate (HDI) - and these are best avoided.
- **Toxic catalysts:** minutiae amounts of harmful organomercury salts, such as phenylmercuric neodecanoate, were found to be excellent, highly selective catalysts for polyurethanes; less concerning but still somewhat nasty tin(IV) compounds (e.g., dibutyltin dilaurate) offered some hard-to-replicate benefits, too. Needless to say, you probably want to avoid unnecessary risks. Such catalysts are largely phased out in favor of compounds of zinc, bismuth, tin(II), titanium, aluminum, and so on - but you can occasionally find them in products available in some markets (for example, Smooth-On, BJB, Alumilite, and Freeman still use them frequently).
- **Possibly harmful amines:** several amine curatives once-popular in certain high-performance elastomers - most notably 4,4'-methylenebis(2-chloroaniline) (MBOCA / MOCA) - were implicated as probably carcinogenic in humans. Be watchful for products that haven't switched to safer, modern alternatives.

You won't find any of the problematic components in resins from Innovative Polymers - but other manufacturers sometimes show less restraint. Request and study material safety datasheets (MSDS) when in doubt.

#### 4.1.4. Flexible polyurethanes

Flexible polyurethane elastomers are an interesting alternative to silicones. You don't necessarily need to buy any, but they may come handy if you wish to make functional rubber parts, and your silicone is not pigmentable, or has insufficient strength.

Key advantages of these rubbers include 30-50% lower price, and much better performance toward the upper end of Shore A (above 40 or so): several times higher tear strength, much less pronounced tear propagation, excellent abrasion resistance, lower coefficient of thermal expansion, lower viscosity, and the ability to pigment the system as seen fit. On the flip side, polyurethanes exhibit some exotherm-caused shrinkage in larger castings, and adhere to many other plastics, making them less desirable for creating negative molds; they also tend to have slightly worse rebound characteristics, and limited temperature resistance (they get soft around 70-90° C, and deteriorate

somewhere between 150 and 200° C - so casting low-melt metals, for example, is out of question).

The parameters to look for in these compositions are similar to these for silicones, with the exception of pot life and viscosity - here, the advice provided for polyurethanes is more pertinent. When reviewing the datasheets, pay close attention to the ratio of tensile strength to elongation at break, because there can be striking differences in the rigidity of various products, especially those rated 70 Shore A or above; some of them are only somewhat flexible, and will not be suitable for making parts such as tires, rollers, or transmission belts.

**Note:** This goes both ways: not all Shore A polyurethanes are particularly rubbery, and not all Shore D polyurethanes are necessarily very rigid. For example, some 60 Shore D polymers are highly elastic and can be stretched up to three times their original length, even though their surface feels hard as nails. They resemble the rubbers used in certain garden hoses, shopping cart wheels, etc.

As for recommendations: you should go with Innovative Polymers HP-21xx series; I've tried quite a few other products, and nothing else comes close. They are relatively inexpensive (\$25 per liter) and feature superb "true rubber" mechanical characteristics, long pot life, and good cure profiles. For example, HP-2170A is a super-stretchy 70 Shore A rubber with tear strength of 42 kN/m, far surpassing most silicones. There are also softer variants, down to 50 Shore A (HP-2150A); and more rigid but still surprisingly flexible ones, up to around 60 Shore D (HP-2160D). In fact, it's possible to blend them to achieve intermediate properties as needed for a particular project.

All the products in the HP-21xx line take several days to polymerize at room temperature; if that's too slow and heating them up is not an option, you can add some separately purchased catalyst to get overnight cure with no real trade-offs.

#### 4.1.5. Epoxies, polyesters, and so on

If you have dabbled in resin casting before, chances are, you used epoxy or styrene-based polyester resins, rather than polyurethanes. These options are popular with hobbyists because of their broad availability, low price, and less onerous processing requirements (i.e., less sensitivity to mixing ratios, moisture contamination, etc). That said, I think it's a bad idea to use these materials in precision casting work, for a couple of reasons:

- **High shrinkage.** Both polyesters and epoxies cure by condensation, and usually exhibit very high exotherm compared to polyurethanes with the same demold time. Shrinkage as high as 10% is common in unfilled polyesters; epoxies are better, but not dramatically so.
- **Mold damage.** Epoxies and polyesters use corrosive, oxidizing, or permeating components, and this reduces the service life of silicone molds quite a bit. It may be difficult to get 5-10 pulls out of a complex mold.
- **Brittleness.** Most epoxies and polyesters are designed to be cheap and to bond well, and not to have outstanding flexural parameters or impact strength. When superior characteristics are required, fiberglass, carbon fiber, or aramid cloth is typically used for reinforcement, but that's not very feasible for tiny, complex parts.
- **Low UV resistance and clarity.** Transparent epoxies and polyesters usually have lower clarity than water-clear polyurethanes, and are more prone to developing a yellow hue when exposed to sunlight. In comparison, many polyurethanes exhibit high UV stability.
- **Workshop safety concerns.** Components of a polyester resin are extremely volatile, flammable, and are an inhalation risk. Epoxies are more user-friendly, but use corrosive chemicals, and contain non-trivial amounts of bisphenol A. Bisphenol A hasn't been conclusively implicated in being harmful to humans, but is suspected of being an endocrine disruptor based on some subset of animal studies. Limiting your non-essential exposure may be a wise step.

Of course, I don't want to demonize these polymers. Polyesters are sometimes useful for bonding and repair applications; and epoxies are extremely useful as high-performance glues and laminating or potting resins. Both epoxies and polyesters can be superior in lay-up composite applications, too, in part owing to their improved bonding capabilities. You can have a look at the products sold by [Freeman Supply](#), or check out the low-cost,

water-clear epoxies available from [Polymer Composites Inc](#); just don't expect them to be a sensible match for PU for the processes discussed in this guide.

#### 4.1.6. Pigments and dyes

Adding colors to your castings is fairly easy. One option is to find an artist store, and shop for dry, non-toxic organic pigments. Such pigments will work equally well with polyurethanes, silicones, and just about anything else - but may be relatively painful to disperse or to blend with any accuracy. The other choice is to purchase coloring pastes where pigments are already dispersed in a non-reactive (plasticizer) or reactive (polyol) medium - but these won't work in silicones.

For dry pigments, [Kremer](#) is probably the best source online. For reactive dispersions, you can ping Innovative Polymers (send them your picks from [this RAL chart](#)); [Eager Plastics](#) has a pretty good selection of non-reactive pigments and dyes, too.

Beyond that... selecting your palette is a matter of personal preferences, but here are some quick tips:

- Be sure to get some titanium-based white (Kremer 46200, IPI Bright White) and deep black (Kremer 47400, IPI Black 201). These two are used not only as standalone pigments, but are also useful as opacifiers for controlling transparency of the material, and the saturation produced by other dyes.
- When browsing artist stores, try to avoid most mineral-, plant-, and animal-based pigments; stick to modern synthetic dyes where possible. In particular, be wary of traditional products containing lead, mercury, arsenic, cadmium, chromium, and nickel.
- It's easy to go from vivid hues to muted ones, but not the other way round. Focus on punchy pigments with simple names; "Bohemian moonlight blue" is not what you need.
- You can't just buy three primary colors, and hope to make every other hue with that. It won't work, especially if you are aiming for a vibrant palette. Aim to get at least 6-8 different, useful colors to begin with.
- Whatever you do, keep your pigments moisture-free. Store them in closed containers and in a dry place.

If you need inspiration, here are some nice picks:

- **Daylight fluorescent colors:** Kremer has a [line of day-glo pigments](#) that result in extremely vivid colors that form translucent dispersions; Innovative Polymers carries a good selection, too.
- **Basic opaque pigments:** in addition to black and white, you may want to start with vibrant yellow (Kremer 55100, IPI Yellow 802), cyan (Kremer 55500, IPI Blue 303), pure red (Kremer 55300, IPI Bright Red), magenta (Kremer 55470, IPI Magenta), green (Kremer 55700, IPI Signal green), orange (Kremer 55200, IPI Orange 602).
- **Transparent dyes:** useful only if you are planning to make transparent or translucent parts. [ORASOL dyes](#) produce brilliant transparent colors, but [pre-made dispersions](#) from Eager Plastics are more convenient to use and have very high yields.
- **Fancy pearlescent pigments:** check out [Paint With Pearl](#). Their prices are hard to beat.

A small bag or bottle of a pigment will cost somewhere between \$5 and \$20, and should last for years.

***Tip:*** You can, of course, opt to paint your parts instead of adding pigments to the resin; acrylic and polyurethane lacquers can be used alike. For adding text or other ornamental elements to machined parts, you may also want to consider a low-cost vinyl cutter, such as Silhouette SD or Roland Stika SV-8. These devices are fairly affordable, and the results look [amazingly good](#) - especially if a layer of [clear coat](#) is applied on top.

If another cutting machine is too much, you can also simply equip your CNC mill with a specialized [drag knife](#) to get comparable results.

#### 4.1.7. Other additives

Adding colors aside, many other properties of cured plastics can be altered in profound ways by introducing certain easily available, low-cost additives. It's almost impossible to provide an exhaustive overview of all the available choices, but several use cases are definitely worth calling out:

- **If you want to reduce shrinkage,** there are many fillers that increase the distance between reactive molecules of the resin, slowing down the reaction and buffering some of the produced heat - but the ones that do so without significantly affecting viscosity or mechanical properties of the finished part are of particular interest.

I found that [3M Scotchlite iM30K](#), a type of high-strength, hollow glass microspheres (less than 0.02 mm in diameter) works great when added at a ratio of about 4% by weight; when used in conjunction with the pre-mixing trick discussed earlier on, shrinkage is eliminated almost completely. When flexural properties are of utmost importance, tiny [milled glass fibers](#), around 0.8 mm long, significantly reduce shrinkage at a ratio between 10 and 30% by weight. Both of these fillers sell for less than \$20 per liter.

- **When you need to reduce weight,** [Scotchlite S15](#) can be added to any system at ratios as high as 3:1 by volume, resulting in a free-flowing resin with effective density less than 0.4 g/cm<sup>3</sup>. Compressive strength will be pretty reasonable, but flexural properties will suffer. This may come handy especially for hulls of model airplanes or boats.
- **When you want an extremely hard plastic,** the aforementioned variety of short glass fibers can substantially improve flexural strength, more than double flexural modulus, and significantly increase hardness and abrasion resistance - especially when used in conjunction with a silane coupling agent (see later on). Using longer or thicker fibers is a possibility, but may affect surface finish or complicate the casting process. For short fibers, ratios around 30% by weight are a good starting point. The downside of this trick is a significant reduction in impact strength; starting with a high-impact resin, such as TP-4052, may give you a bit more wiggle room.

- **For improved impact strength, lower hardness, or lower viscosity,** plasticizers can be introduced to the system to effectively add "padding" in the resulting polymer structure. Because they don't participate in the polymerization reaction, they reduce tensile strength and may increase creep (the tendency for the material to permanently deform under relatively minor but sustained load), but that trade-off is acceptable in many uses.

For polyurethanes, one of the safer and more versatile plasticizers is dipropylene glycol dibenzoate (DPGDB), available from [Eager Plastics](#) under the name of EP9009. For silicones, any low-viscosity silicone oil will do; you can get it from the same source that you are purchasing the resin from. Plasticizer levels between 5 and 10% are common, although in silicone rubbers, up to 50% may be useful. You shouldn't have to pay more than \$20-\$40 per liter or so.

- **To simulate select properties of metals:** powdered metals, such as copper, aluminum, or iron, can be added to resins in significant amounts to increase their weight, thermal conductivity, and remarkably, give them a highly metallic appearance and feel when polished. You can find suitable 300-600 mesh metal powders on eBay or from sources such as [ArtMolds](#). Prices vary, but aren't prohibitive. Flexural properties will suffer.

Of course, multiple types of fillers can be combined; in particular, it may be useful to add some glass fibers to lightweight materials filled with hollow glass spheres, to maintain acceptable flexural strength. If you are wondering what to buy up front, it's not a completely bad plan to get some Scotchlite iM30K, plus 0.8 mm glass fibers, and a bit of DPGDB (also useful for preparing pigment dispersions and such). Other fillers are not nearly as essential.

#### 4.2. Casting workshop

Now that we have the selection of resins, pigments, and fillers sorted out, it's time to briefly chat about the

workshop equipment you will need to get the ball rolling. The list isn't particularly long, but even when it comes to something as inconsequential as mixing cups, making the wrong choice will unnecessarily complicate your life.

#### 4.2.1. Vacuum pump and chamber

---

Insufficient mixing may cause a range of problems with finished parts - but vigorous stirring will almost always introduce some air into the resin. This problem aside, bubbles of air may get trapped inside mold crevices as you pour the mixture in - even if your mixing skills are beyond reproach. Last but not least, some resins may simply liberate some amount of dissolved gases once the polymerization reaction begins; IE-3075 is an example of that. Vacuum degassing solves all these problems, and is not as scary as it may sound.

Even if you are on a tight budget, you should get a low-cost vacuum pump capable of getting pulling around -1000 mbar of vacuum (that's about 10 mbar absolute, or -29.5 inches of mercury). I ordered mine from [VIOT: click here](#) for an entry-level model that sells for around \$100, and should work fine; I have a \$140 model with a higher flow rate, and it served me well. You will also need some sort of a vacuum chamber to hold the mixing container and the mold itself; it's possible to rig something together on your own, for example using a sturdy cooking pot and a cover made out of thick polycarbonate - but [low-cost vacuum dessicator chambers](#) work fine in that capacity, and start around \$50 or so. If you are planning to work on relatively small projects, or opted for a pump with a relatively low flow rate, Bel-Art #420100000 is a good choice; otherwise, model #420430000 will accommodate larger pieces, too.

(For a bit more money, you can also find some good-looking purpose-built degassing chambers from [several sellers on eBay](#).)

About the only other piece of this puzzle is a hose to connect the pump with the chamber. I use 1/4" Kuriyama K7160 Polyspring, which I ordered from [these guys](#); it's about \$4 per meter. Other hoses can be used, too, but they need to be vacuum-rated, which in practice means that they need to be either fairly rigid, or reinforced with a metal spring to avoid collapsing as soon as you turn the pump on. Beyond that... well, getting some vacuum grease for o-rings and other parts of the vacuum chamber is not a bad plan ([link](#)); it also comes handy for several other purposes every now and then - for example, for preventing caps on resin bottles from seizing in storage.

#### 4.2.2. Auxiliary tools

---

It goes without saying that for any sort of precision work, you need a reasonably well-equipped workshop to begin with. I am assuming that you already have that - and that basic tools such as clamps, needle files, or several grades of fine sanding paper, are always within reach.

With that in mind, there are several less obvious, minor items that will be useful specifically when casting mechanical parts, and that we haven't mentioned before:

- **Precision scale:** observing proper mix ratios is extremely important for addition cure silicones - errors of as little as 2% may affect important properties of the rubber. Polyurethanes are a bit more tolerant, but it is important to stay within 4% or so.

To that effect, in order to accurately mix components when working with relatively small quantities of resins (10-50 ml), you should try to get a scale with a resolution of at least 0.05 g, and measurement range of at least 0-200 g. You can get something usable for under \$40 (I ordered from [wholesale-scales.com](#), and can't complain). If you are watching your budget, you can also get away with a regular, gram-resolution kitchen scale - but in this case, you shouldn't attempt to mix less than about 30 ml of polyurethane, or 100 ml of silicone. In the long haul, this may be wasteful, depending on the size of your parts.

- **Mixing containers:** any plastic cup should do, but consider getting a box of one hundred 100 ml polypropylene beakers from [Ted Pella](#) or any other outlet with lab supplies; the price should be around 20 cents a piece (\$20 per box). PP beakers are nearly indestructible, will not crack during mixing, will not melt when heated up, and will not fall apart when exposed to solvents. But perhaps most importantly, resins don't stick to them, so you can reuse the same cup over and over again. This actually leads to savings after a month or two of casting work!

- **Mixing sticks:** you can mix the resin with just about anything, as long as it's dry, clean, and has the

right shape to reach all the spots inside the mixing cup. Many people use tongue depressors, which cost something around 1-2 cents a piece ([try Ted Pella](#)), and can be wiped clean and reused indefinitely. If you choose that option, just keep them away from moisture, and be sure to clip the business end to get a flat tip for better reach. Tongue depressors aside, chopsticks also work well, as do proper laboratory spatulas and flattened mixing sticks ([link](#)). The choice is yours.

- **Freezer paper or thick aluminum foil:** it's convenient to pour a bit extra resin into the mold, and let it overflow before sealing everything; this minimizes the risk of air entrapment. Any non-permeable, cheap material can be used to protect the surrounding working area against the overflowing resin; freezer paper and heavy-duty aluminum foil from a grocery store will do the trick.
- **Deburring tools:** there are several inexpensive hand tools that prove very useful for quickly cleaning up the thin film of resin that may be present around the mold's parting line, and for fixing other minor casting artifacts. I recommend grabbing one miniature swivel deburrer, such as Noga RB1000 (\$8); one fixed-blade scraper, such as Noga SC8000 (\$20); and perhaps a couple of small, hard Nylon brushes, such as [this kit](#) (already mentioned before). You can do without them, and just use a toothbrush, a pin, and a needle file - but your life will be a bit more difficult.
- **Dental picks / wax carvers:** a low-cost set of picks, hooks, and spatulas - something like [this kit](#) (\$9) - will be extremely useful for removing silicone molds from masters, and for assorted other precision work. In fact, if you don't have that in your workshop already, you are missing out.
- **Sheet polypropylene or HDPE:** flat, smooth, non-stick plastics are useful for covering single-part molds to ensure dimensionally accurate and aesthetic castings. Polypropylene is the best choice, and you can get it in cut-to-size pieces from many sources, including [Professional Plastics](#); HDPE is a second option, but it is less temperature-resistant, and it's more common to see warped sheets. In any case, get at least 15-20 cut-to-size pieces, around 1-3 mm thick, with dimensions corresponding to the scale of parts you are expecting to routinely make (10 x 15 cm is probably a good starting point). You can get individual larger pieces later on.

*Note: make sure that the material you are buying is machined to a flat, smooth surface; HDPE in particular is sometimes available in patterned varieties - suitable for making cutting boards, but not so much for casting work. The material may come with some minor scratches, but these can be buffed out easily with fine sanding paper. If the pieces are badly gouged or warped, however, request a replacement!*

- **Brushes:** a set of small, soft-bristled brushes ([example](#)) will be useful for cleaning master patterns and applying mold releases and other treatments to molds.

That's about it! For working with small quantities of resin and pouring it into complex multi-part molds, you may also find it useful to get a box of single-use, two-element 10 ml syringes with no rubber seals (e.g., from [eNasco](#)), and some blunt-tip [dispensing needles](#).

#### 4.2.3. Mold releases and other useful chemicals

Mold release is a material that forms a protective barrier between the mold and the resin you will be pouring in, and makes it easier to demold final parts. The use of mold releases is optional when casting silicones in polyurethane patterns, or polyurethanes in silicone molds, because these materials don't adhere to each other in a particularly strong way - but still, a properly selected release agent makes demolding easier, and prolongs the life of any mold by reducing its exposure to reactive chemicals. On top of that, if you ever want to cast polyurethanes in polyurethane molds, or silicones in silicone molds, a robust adhesion barrier is simply a must.

I have tried many different mold releases over the years, and my top pick is, unquestionably, [Stoner A324](#); this spray-on agent beats most of the silicone, PTFE, or zinc stearate releases that I have tried before. They sell it for about \$5 per can, and this should last you for about 1-3 months of hobby work. Stoner ships for free if you order a full box of 12 cans (\$60).

If you can't obtain this product in your market, silicone-based mold releases are your second best choice, although

some varieties may gradually permeate and swell silicone molds. Releases based on mineral oils, PTFE, zinc stearate, polyvinyl alcohol, and so on, usually don't perform well in high-precision, vacuum-assisted casting work. The manufacturer of your silicone rubbers may be able to recommend some specific, locally available picks.

That aside, you may want to also grab a wax-based mold release: they dry to form a hard, polishable, permanent layer that not only serves as a barrier, but should improve the surface aspect of polyurethane patterns. Low-viscosity brush-on formulations, such as [Synlube 531](#), should work well; more viscous liquids, such as [AdTech MR-1](#), may need to be diluted with naphtha when working with intricate shapes, but it's not a hard thing to do. Hardware store paste waxes that contain carnauba wax ([link](#)) work fine for simple shapes where dimensional accuracy is not critical, but high-gloss finish is desired - and if you dilute them to a water-like consistency, you can use them for complex models, too. In all cases, once the wax is dry, you can simply buff it with a soft cloth or a brush.

**Note:** waxes should not be applied to silicone molds, because they will inevitably crack and peel off - and the solvents used are usually damaging to silicone rubbers, too. In general, with any new mold release, always test it by soaking a piece of silicone in it for about 5-10 minutes; any substantial or permanent change in dimensions would be a reason for concern.

Mold releases aside, you may find it useful to get some of the following, largely optional chemicals:

- **Basic solvents:** it's good to have the following two options at hand:
  - **Acetone:** when used sparingly, works as a great degreaser for plastics; if you don't make a habit of using it, you will not get robust bonds with cyanoacrylate or epoxy glues, especially on molded parts that may be still covered with release agents or so. It is also useful as a solvent for certain resin additives, and helps degas stubborn or high-viscosity resins (1-2 drops make a huge difference). It is advisable to get reagent-grade acetone from eBay ([link](#)); hardware-store variety is usually also OK, but may cause problems if it contains too much water.
  - **VM&P naphtha:** useful for diluting mold releases and for cleaning up silicone spills. Ethyl acetate (an acetone-like solvent used in nail polish) works for the last two purposes, too - and may be a more environmentally friendly alternative.
- **Siloxane surfactants (very optional):** concentrated siloxane surfactants make it easier to use polyurethane resins as surface coatings; they do that improving wetting characteristics and eliminating pinhole defects. You will likely need them if you wish to use resins such as HP-21xx or IE-3075 as high-performance surface coats. Look for a product called Satur~8 on eBay; when added to resin at around 1.5%, it should make quite a difference.
- **Silane couplers (ditto):** these may come handy if you wish to add glass-based fillers to polyurethane resins, or improve bonding to glass surfaces. They modify the surface of these materials to make them form strong, covalent bonds with polyurethanes, epoxies, and similar plastics. The coupler is applied as a 2% solution in alcohol or acetone, and then left to dry in the presence of atmospheric moisture (water molecules are needed for the actual activation process to occur).

The most common and useful compounds are called 3-glycidoxypropyltrimethoxysilane (GLYMO) and (3-aminopropyl)-triethoxysilane (APTES). Concentrated variants can be ordered from [Chemsavers.com](#) (just drop them a mail), or get a smaller quantity from SPI Supplies ([link for GLYMO](#), [link for APTES](#)). Diluted APTES is also available from [crafts stores](#).

Note that you don't always have to go through all this trouble; for example, milled glass fibers available from Fibreglast are already pre-treated in this manner, and there is no point in soaking them in a coupling agent again. On the other hand, Scotchlite glass microspheres come in a virgin state and will benefit from this step.

#### 4.2.4. Optional: temperature-controlled oven

Temperature-controlled ovens are not essential in casting work, but they let you perform several time-saving tasks:

- Drying molds, pigments, fillers, and other supplements if necessary. In most cases, this can be also accomplished with any cheap, \$40 toaster oven.
- Accelerating the cure of resins when dimensional accuracy is not critical. This is sometimes preferred to chemical acceleration, because there is no need to purchase any reagents, and the behavior can be fine-tuned more easily. Thermal acceleration requires accurate temperature control between around 30-70° C, and will usually let you demold parts in under 1 hour.
- Post-curing finished parts, so that they achieve their ultimate properties sooner than normal (it may take up to a week at room temperature for all the reactions to cease). This requires accurate temperature control around 80-120° C.

Low-cost toaster ovens are the obvious solution, but somewhat unfortunately, they have extremely poor temperature accuracy and stability, especially on the lower end of the scale. If you are an accomplished DIYer, you may be able to grab one and simply equip it with a more accurate, digital temperature control circuit. That said, if you are willing to part with a few hundred bucks and have some floor space, you can also try getting a real laboratory oven, such as [this one](#) (\$400); hot air sterilizers and dryers may offer a smaller and slightly more affordable alternatives in some markets, too ([example](#), around \$300).

If you don't have that much money, or simply don't have enough room, don't worry; you can do just fine without this piece of equipment, provided you are willing to wait a bit longer for your castings every now and then.

#### 4.2.5. Optional: pressure pot

There are some complex, multi-part molds where it may be hard to consistently avoid air entrapment, even with the aid of vacuum; on top of that, there are some resins that tend to be difficult to fully degas, or that will develop bubbles of carbon dioxide when not mixed perfectly well, or when exposed to residual humidity. That latter set of problems is particularly evident in mercury-free water clear polyurethanes, such as the vanilla version of Innovative Polymers OC-7086.

To improve your odds when dealing with such tricky cases, it helps to have a pressure pot; the idea is to increase ambient pressure surrounding the mold to about 3-4 bar, thus crushing and dissolving back any existing bubbles, and discouraging the formation of new ones. Sure, it's a brute-force solution - but can you argue with its results?

Pressure casting equipment is more bulky, more expensive, and somewhat more dangerous than vacuum pumps (due to much higher pressure differentials) - and for most part, isn't necessary; get it only if you have plenty of room, and you are either forced to work with water clear resins that don't tolerate non-pressurized casting, or you are willing to spend at least \$200 to improve your yields a tiny bit (say, from 85% to 97%).

If you want to take this route, pressure pots start around \$80 ([link](#)); nicer ones fetch as much as \$300 ([link](#)). To operate it, you will also need a compressor - and these start from \$100 from fairly noisy units from the hardware store, to \$180 for [relatively quiet-running ones](#); standard pressure hoses and fittings add another \$20 or so.

### 4.3. Your first casting projects

All right, all right - enough with all the theory and shopping tips. It's time to dig out that wax pattern that you have made before, and turn it into a finished work of art! This section is all-text, but if you need visual aids, [this photolog](#) is probably good to look at.

**Caution:** similarly to many workshop and household chemicals, casting resins can be harmful if misused. In particular, they may react violently when mixed with incompatible substances; cause severe irritation or lasting damage if splashed into your eyes; and will emit dangerous vapors if overheated, burned, or intentionally aerosolized.

Please refer to product safety datasheets (MSDSes), and to [section 7](#) of this guide, for an overview of

*known risks, material handling recommendations, and disposal procedures. Do not proceed with any hands-on experiments until you have done so.*

#### 4.3.1. Making a silicone mold

Resin casting is fun, but you need to remember that once the components are mixed, the reaction will proceed no matter what; it's important to plan accordingly: read this section fully and memorize all the steps beforehand, and have all the necessary supplies and information within reach.

First, you need to estimate the amount of resin you will need: for CNC-machined or 3D-printed shapes, simply ask your CAD application to calculate the volume of the master, and subtract this value from the volume of a dummy box of the same height, width, and depth. To have a comfortable margin, add about 15% to the result, or 10 ml, whichever is greater. Then, multiply that volume by the density of the resin (check the datasheet; it's usually between 1.1 and 1.4 g/cm<sup>3</sup>). The resulting figure is the weight of the material you need to prepare.

With these calculations out of the way, prepare the following stuff:

- A roll of paper towels. Tear away about 5-10 pieces ahead of time, and place them within reach; this comes handy for wiping mixing tools, cleaning up minor spills, etc.
- An easily accessible trash bag for dirty towels and other things you want to get rid of in a hurry.
- Mixing container that fits within your degassing chamber, and holds at least 3 times the volume of resin you will be working with.
- A clean, flat, and clutter-free work surface, covered with aluminum foil or freezer paper.
- A kitchen timer, a stopwatch, or something else that would let you keep track of time.

Although accidents are unlikely, you should still try to minimize potential damage: if there are any LCDs or other expensive gizmos in your workshop, consider moving them a bit farther away, or covering them with plastic sheeting. Don't wear your best clothes, and if you have a carpeted floor or expensive hardwood, cover the area that is most likely to suffer in case of a spill.

When you are ready to go, place the mixing cup on a scale, then tare it. Agitate both components of the resin in their original containers, and then use a clean tongue depressor or a spoon to pour about 20 g of silicone into the mixing cup. Add a suitable amount of catalyst (the ratio is usually 10:1, but check the datasheet), start the timer, and begin mixing thoroughly for honest 3 minutes; be sure to repeatedly scrape the sides and the bottom to avoid leaving any unmixed resin in these spots.

Next, place the cup under vacuum; the mix will initially rise as the bubbles expand, and then collapse back; you should keep it under vacuum for another 1-2 minutes past that point. If the resin gets dangerously close to overflowing during the initial rise, simply release the vacuum (i.e., yank out the hose), and try again; several cycles of that should do the trick - and next time, use a larger cup.

The next step is to pour some of the resin into the mold created in chapter 3; use about half of the required volume or so. Place the mold on a strip of aluminum foil, and put that under vacuum; this will help the resin conform to even the most intricate shapes with no effort on your end. After about 2-3 minutes, you can slowly release the vacuum, pour the remaining amount of resin (use a bit more than necessary to get a convex surface at the top of the mold), and set the whole thing aside for several minutes to allow any bubbles to rise to the surface (or collapse back into the resin). If there are any stubborn bubbles on the surface at that point, you can apply a gentle burst of compressed air to get rid of them.

Finally, cover the entire contraption with a flat, clean sheet of polypropylene; lay it down gradually, starting at one side, to avoid air entrapment. When done, weigh it down with something reasonably heavy - around 500 g should do - and brace the whole thing, so that the cover doesn't slide off. Be sure to check the timer at this point - has the entire process taken more than about 12-15 minutes? If yes, why?

In any case, give it about 12 hours or so, until the resin remaining in the mixing cup is firm and tack-free. If you are impatient, placing the mold in a temperature-controlled oven at around 50° C will cut the curing time down to 1-2

hours or so - but don't go too high, given that this particular master is made out of wax. When the rubber is ready to demold, pull off the cover, and then use a dental hook or a similar tool to pry the rubber off near the corner of the mold. Remove it fully and inspect the result. If it looks flawless - as it should - you may want to briefly post-cure it at around 100° C for 30-60 minutes, and in the meantime, pat yourself on the back!

Here are several questions that may be on your mind:

- **What about the application of a mold release?** Waxes already have a non-stick surface, so if you are following the instructions from chapter 3, you should be fine. When using prototyping boards or 3D-printed materials, you should apply a thin layer of mold release beforehand, and allow it to dry. Use a soft brush to spread the substance evenly, and don't overdo it: you want to wet the workpiece, but do not let the liquid pool in hard-to-reach spots.
- **The resin didn't cure! What now?** Ow, bummer. The most likely reason is insufficient mixing (especially if there are sticky patches or streaks in an otherwise well-cured material); bad mixing proportions; contact with incompatible materials (platinum-cure silicones are particularly sensitive to sulfur compounds); or low temperature (the resin will take forever to cure below 18° C or so). If you have ruled these causes out, feel free to drop me a mail, and we'll try to get to the bottom of it.
- **What if I'm having trouble demolding?** This takes some practice - try to find the right tools, and locate a spot where you can grab the rubber with a hook without causing any damage. For particularly challenging shapes, introducing pull tabs and draft angles in the mold, as discussed earlier, is the way to go.
- **What if the mold is too big to put under vacuum?** For simple shapes, you can just gently pour the degassed rubber in, starting at the lowest point of the mold. If there are any tight spots, it's useful to fill a syringe with a small amount of catalyzed resin, put a needle on, and use that to force the resin into these problem areas beforehand.

When working with truly oversize pieces, you may find it useful to rely on a process called *glove molding*: brush on several thin layers of silicone onto the part, and then coat this flexible skin with a rigid, thixotropic polyurethane, epoxy, or polyester resin to create a backing shell. This is a great way to conserve silicone, although the mold takes more time and effort to prepare.

(For optimal results, the silicone resin used for this purpose may need to be thickened to a non-flowable consistency with an additive such as Bluestar 22646 or with fumed silica. Fumed silica can be also used as a thickener for the rigid resin used as the outer coat.)

- **Vacuum degassing only made the bubbles worse, what now?!** The resin should rise and collapse pretty quickly; some bubbles will continue to appear, but they should collapse back shortly after the system is returned to room pressure. If that doesn't happen, check the ballast valve on your pump (it should be closed) and examine the entire setup for any leaks. Make sure that the pump is in good working order (many pumps require the right amount and type of mineral oil for lubrication). Verify that the pump is well-suited for the size of your vacuum chamber, too. If you're stumped, getting a cheap vacuum gauge from Amazon may help.
- **How long will the mold last?** Depends. When following the advice provided in this guide, around 50 pulls would be the norm. Lower-strength silicones, more aggressive resins (epoxies, polyesters, very fast or very slow polyurethanes), and complex geometries (deep molds with intricate detail) will lower the lifespan of silicone molds.

Okay, okay - time for some polyurethane fun!

#### 4.3.2. Casting plastic parts

Polyurethane casting is not dramatically different from working with silicones, but you have to be swift, and pay more attention to detail. Quite simply, with a resin such as IE-3075 or TP-4052, you will probably only have about 6 minutes to go through all the steps - so there is no time to look around for paper towels or a mixing stick.

In general, before starting, you should go through all the preparation steps outlined in the previous section; and

when done, you should also do the following:

- Make sure that the resin hasn't separated or crystallized. In almost all formulations, the isocyanate component should be clear, and both components should be completely homogenous. Agitate the containers if necessary; crystallization may require heating the containers slightly (usually not more than 40° C); always check with the manufacturer first, and exercise caution.

**Note:** *in filled systems, the filler settles out of the solution and hard-packs at the bottom of the container; this is particularly true for heavy fillers, such as glass. You need to use a [small paint mixer](#) attached to a power drill to properly homogenize the contents of larger containers; it's not a big deal, but watch out for spills. Manual mixing is viable only for small quantities, up to perhaps 250 ml or so.*

- If you haven't done so already, get two smaller polypropylene bottles, and fill them up with a more manageable amount of isocyanate and polyol, so that you do not have to constantly agitate, open, and tip over gallon-size jugs. When done, blanket the original containers with "canned air" or nitrogen, and close them tightly. Be sure to label the new bottles clearly, too.
- Make sure that the mold and any hygroscopic fillers you want to add are perfectly dry - especially before first use, or after extended storage. Placing solid materials at 110-150° C for about 10-20 minutes, or applying vacuum for 5-10 minutes, is a good option if you want to play it very safe.
- Prepare a perfectly flat, spill-proof surface on which the silicone molds will be laid out while the resin cures. You can use a sheet of polypropylene, or a neat piece of aluminum foil with no creases. Whichever option you choose, applying some mold release to that surface is a good idea.
- Grab a small, flat sheet of polypropylene or HDPE to serve as a top cover for the mold. Mark one side with a permanent marker, apply a thin coat of mold release to the other, and place the sheet within reach.
- Apply a very light coat of mold release to the silicone mold, and spread it evenly with a brush. Repeat the process every 5-10 pulls, but don't overdo it; for casting polyurethanes in silicone molds, too little is better than too much. Allow the release to dry for 15 minutes or so.

With all the preparations taken care of, you are ready to roll. Place the mixing cup on the scale, tare it, and pour the required amount of isocyanate; in our case, 10 g will do. Next, gently pour the appropriate amount of polyol (for IE-3075, this will be 8.9 g), start the timer, and begin mixing very thoroughly, frequently scraping the walls. Most polyurethane resins, IE-3075 included, require at least 90 seconds of mixing to progress from a dispersion to a proper solution when mixed in a small quantity - and if you stop sooner, the cured material may not cure properly. Do it right.

When done, place the container under vacuum, and turn on the pump. The resin should take no more than about a minute to rise and collapse back; if it's taking much longer, your vacuum chamber may be too big, or the pump may be inadequate or malfunctioning (e.g., due to an open gas ballast valve, or due to contamination). If, on the other hand, everything went as expected - and the bubbles have collapsed in a timely manner - you should now pour some of the resin into the mold cavity (to fill it roughly halfway, taking care to cover any detail where air entrapment is likely), and place the mold under vacuum for another minute or so. Don't worry if it never stops bubbling at this stage; that's OK.

After a brief round of degassing, release the vacuum gently, place the mold on a previously prepared flat surface, and add the remaining resin, until it overflows and forms a convex surface (this is important - otherwise, air entrapment is a lot harder to avoid). Grab the polypropylene cover and carefully lay it on top of the mold, using the technique outlined for silicones. You should brace the cover against something, so that it doesn't slide off, and weigh it down with around 200 g (larger molds can be clamped with several kg of force, but this particular one is relatively easy to squish).

Well, that's it! Before you go, check the timer; if the process has taken more than six minutes, you should figure out how to improve it. In any case, leave the mold alone for at least 3 hours (or more, depending on the resin used), and when you come back, confirm that the material left over in the mixing cup is tack-free and hard as nails. Next,

gently flex the polypropylene cover to detach it from the part, and extract your casting from the mold.

Hopefully, the result is perfectly fine; that said, the likelihood of mishaps is higher for polyurethanes than it is for silicones - so if something isn't right, don't despair. Here's a quick summary of the most common issues I have seen:

- **The resin is still sticky or gooey:** assuming you waited long enough, the culprit may be low ambient temperature (try to stay close to 20-25° C), incorrect mixing proportions, or mixing the wrong components (the containers all look almost the same). A less likely cause may be the introduction of a problematic filler or dye, or the use of a resin that is incompatible with platinum cure silicones (this is rare; the datasheet should have said so). Treating the surface of the mold with a separately purchased polyurethane catalyst is a possible workaround for that last problem - see later on.
- **The material turned into a foam:** this may be caused by using a resin that crystallized during storage and hasn't been heated up to dissolve the crystals; by adding a filler or other additive that contained a significant amount of water; or by similar type of gross moisture contamination (wet hands, condensation, etc). In some formulations, failing to properly degas the catalyzed resin, mixing for way too short, or not allowing the mold release to dry properly before pour, may also cause a similar symptom.

(If you suspect your vacuum setup, check out the tips in the previous section.)

- **Top surface is soft and scratches easily:** this is almost always due to insufficient mixing. Dispersed droplets of polyol have coalesced and, being lighter than the isocyanate component, floated up. More thorough mixing or higher ambient temperature will usually fix the issue. Premixing - as discussed later on - should also help.
- **Discoloration or clusters of tiny bubbles running around edges or in other well-defined streaks:** usually also a matter of insufficient mixing - see above.
- **Several individual, round bubbles that floated to the top or are stuck in tight corners at the bottom of the mold:** air entrapment during the casting process. If there are stubborn bubbles still present in the liquid as you are pouring it into the mold, and your vacuum system is in good working order otherwise - mix in 1-2 drops of acetone per 10 ml of resin before degassing, and see if that helps.

Oh - when in-mold degassing is not possible due to the size of the part, applying the resin with a syringe or a brush may help with any tricky spots.

- **Giant tear-shaped bubble near the top surface:** air sucked into the mold due to applying too much force when placing the cover or insufficient clamping. It may also be that you haven't poured enough resin to begin with, or that the cover or your work surface is warped.
- **Several oddball bubbles in thin sections, especially if suspended in the middle of the part:** often caused by using a highly moisture-sensitive and slowly curing resin, such as OC-7086; the bubbles form as the resin is already partly-solid, contributing to their sometimes elongated or meandering shapes. If all other options fail, pre-heating the molds, adding some extra catalyst (see later), or using pressurized casting will help.

If you are seeing any issues, it's important to narrow the problem down right away, while the number of variables is still fairly low. If you are out of ideas, don't hesitate to ping me at [lcamtuf@coredump.cx](mailto:lcamtuf@coredump.cx); I may be able to help. A good place to discuss your experiences or showcase your work may be [/r/resincasting](https://www.reddit.com/r/resincasting), too!

Anyway - if everything turned out to be just fine, you may want to briefly post-cure the part and any leftover material. Place it at around 100° C for one hour, and then play with it to get a sense of its physical properties: try drilling a hole in or scratching the surface of one of the leftover bits, and see how hard it is to break it.

## 4.4. Miscellaneous polyurethane casting tips

This section is just a quick a collection of random notes that should come handy in real-life projects, but that I

wanted to keep out of your first casting job. Enjoy!

#### 4.4.1. Meta: not all resins are alike

Before we dive into various advanced topics, you should know that there are significant differences in the handling characteristics of various polyurethane resins, even if the advertised cure times and ultimate physical properties of the material are roughly the same. In particular, be aware of the following:

- The choice of catalysts will affect the resin's sensitivity to moisture, often dramatically so. Some resins with highly selective catalysts will cure pretty well in open air, and can be used as surface coats (e.g., HP-21xx); some have moderate sensitivity that can be greatly lowered using the premixing technique outlined later in this chapter (IE-3075, TP-4052); and others, such as OC-7086, really need to be modified with an aftermarket catalyst, or pressure-cast, to maintain sanity when making precision parts.
- The use of blocked or temperature-sensitive catalysts helps achieve long pot life followed by a snap cure when casting larger models; unfortunately, the same trick may prolong demold time when working with tiny parts that never reach the activation threshold.
- The exact composition of the system will greatly affect miscibility of the formulation at lower temperatures, before the reaction really kicks in. Some resins will require very little stirring when working with quantities around 10-20 ml, but a few may need as much as 5 minutes of work, unless you are willing to increase the temperature slightly (to ~30° C) or employ a similar trick.
- Depending on the composition, the resin may have a brittle stage, at which point it is very vulnerable to cracking even under minimal stress; in other systems, incomplete polymerization is simply marked by susceptibility to plastic deformation.

Because of all these striking differences, don't take everything you see in this guide as universally applicable to every formulation on the market; and in the same vein, don't expect your own experiences with product A to be fully applicable to product B. If in doubt, request a sample of any new product you are considering, or simply ask.

#### 4.4.2. Adding pigments and fillers to resins

Let's start with something simple. Non-reactive (i.e., plasticizer-based) liquid dyes and coloring pastes can be mixed into the working amount of isocyanate, before adding any polyol; when taking this route, just try to stay under 2% by volume (around 8-10 drops per 10 ml); if you find yourself routinely having to add more, consider switching to dry pigments or a higher-yield dye or a reactive carrier - because past this point, solvents used in the dye will be affecting the properties of the part.

Reactive coloring pastes designed specifically for polyurethanes use a polyol as a base; that's the case for pigments from Innovative Polymers. Their main benefit that they can be added at much higher levels without completely messing up the properties of your parts. In principle, you should subtract the weight of the added dye from the required weight of polyol - but in practice, this varies from one formulation to another. In a quick experiment with IE-3075, I found that using the nominal amount of polyol results in improved strength. Results in other resins may vary.

When working with dry pigments, there is a bit more legwork involved. If you simply dump the pigment unceremoniously into the liquid, it will probably clump together - and stay that way. To avoid this, you need to place the desired amount of material in an empty mixing cup, tare it, and start adding isocyanate drop by drop, mixing constantly, until you end up with a homogenous, runny paste (siloxane surfactants can make the process easier, too). Once the paste looks good, you can gradually add the remaining isocyanate while constantly mixing - and you should be all set.

**Tip:** if you have a high-yield, hard-to-disperse powdered pigment that you keep coming back to, it may make sense to make a custom coloring paste for future use. Simply disperse it thoroughly in an inert plasticizer (e.g., dipropylene glycol dibenzoate, discussed earlier; silicone oil works for platinum cure rubbers) or in a suitable polyol, and pour that into a [dropper bottle](#).

Of course, as noted earlier, you should ensure that the material is moisture-free; in tricky cases, premixing the resin or adding zeolite should help. Glass-based fillers, such as Scotchlite or milled fibers, may benefit from being pre-treated with a silane coupler, too.

#### 4.4.3. Premixing the system

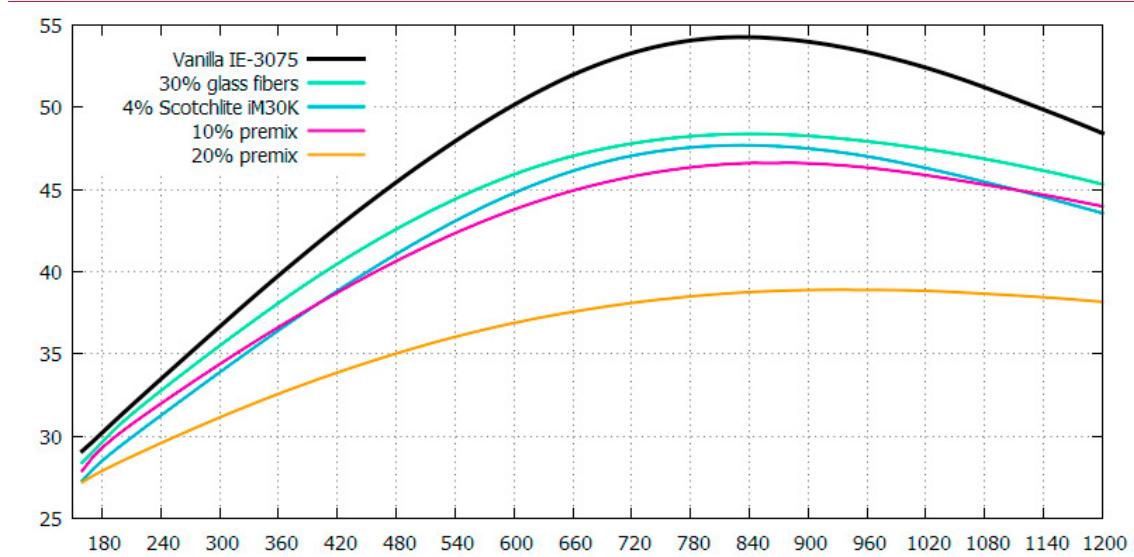
Premixing is one of the simplest and lesser known tricks that can help solve many of the problems that crop up in some polyurethane casting jobs. For example, it can dramatically reduce shrinkage without affecting cure time; lower the risk of cure inhibition; halve the time needed to fully mix the resin in small batches; and greatly reduce the sensitivity to moisture, to the point of making many systems suitable for surface coating applications. These benefits stem from the reduced reactivity and improved compatibility of a partly polymerized liquid. It's not a silver bullet, but for many formulations, it's pretty close to being one.

The only price to pay for premixing is an increase in viscosity, which typically isn't a big deal if you have started with a low-viscosity system such as IE-3075; and the added expense of about 5 minutes of work and about one hour of waiting per every batch that you intend to cast.

If you want to try it out, the recipe is very simple: measure the desired amount of isocyanate, add all the fillers and dyes you want to have, and then introduce between 10% and 20% of the necessary amount of polyol; this mix needs to be stirred thoroughly, degassed - and then stored in a covered cup, blanketed with an inert gas, for about 1-2 hours. At that point, the remaining polyol can be mixed in, and the resin can be cast.

Keep in mind that premixed resins will have short shelf life: the viscosity will keep increasing, and in presence of a catalyst, the isocyanate will more aggressively react with ambient humidity and deteriorate. You should premix only the amount you intend to use right away.

Oh - in case you are curious, this graph shows the impact of premixing on the exotherm for 10 g of IE-3075, and how it compares to the use of fillers. The measurements were taken using a thermocouple submerged in a small, insulated polypropylene cup, approx. 30 mm in diameter:



The X axis is time in seconds. The Y axis is temperature in °C. The resin and the room are initially at around 20 °C.

#### 4.4.4. Adjusting room-temperature cure speed of existing resins

Here's another tidbit you won't find in any other hobbyist reference on resin casting: it is possible, and in fact fairly easy, to chemically slow down many polyurethane systems to significantly reduce shrinkage; and to accelerate slow-curing ones to get your parts sooner or have fewer artifacts in thin sections of your molds. Why bother, you may ask? Well, it not only saves you money, compared to buying several resins for different applications - but perhaps more interestingly, it enables you to come up with custom-tailored cure profiles that are of no commercial interest to the manufacturer.

In essence, there is a wide variety of catalysts used in castable polyurethane resins. Every catalyst behaves differently: some are highly active at room temperature, some kick in only later on, when the resin has warmed up due to exotherm. Some are better at driving the early stages of polymerization, but stop shortly thereafter; some have a sustained effect until the very end. Some are highly selective toward the desirable isocyanate-polyol reaction, and some don't mind catalyzing the isocyanate-water reaction - which leads to the formation of bubbles of CO<sub>2</sub>. Some are very stable, and some deteriorate when exposed to open air and other substances, which may cause inhibition or poor surface cure. But there is no single product that gives you the very best on all fronts.

For this reason, manufacturers combine various catalysts to reach a compromise that makes sense for their intended customers - but these parameters aren't necessarily ideal for your needs. For example, OC-7086 is a resin designed for larger castings; when dealing with tiny parts, it will cure too slowly, and with far too much sensitivity to ambient moisture.

Thankfully, you can fix this on your own.

#### 4.4.4.1. Speeding things up

Ideally, if you wish to use OC-7086 or HP-21xx - or accelerate any other finicky resin - you should get bismuth neodecanoate from [Santa Cruz Biotechnology](#) or [Krackeler](#). The cost is around \$35 for 250 g, and that amount will last you forever. The catalyst isn't dangerous, but both of these places have a blanket policy of shipping only to commercial addresses. If you can't have it shipped to work, ping the folks who run [Chemsavers.com](#) - they should be able to get it for you and ship it to your home for a very modest premium.

Bismuth neodecanoate is a syrupy liquid which needs to be diluted with a plasticizer (e.g., DPGDB), a suitable polyol, acetone, or something else of that sort. Depending on the resins you are working with, you may have to experiment with dosage, but typically, levels between 50 and 500 ppm will be enough. For example, to "fix" OC-7086, you can prepare a 4% solution in plasticizer, and add it at about 1-2 drops per 10 ml of isocyanate as you are getting ready to mix it with a polyol. To speed up HP-21xx, you will need a solution closer to 50%, added in similar quantities.

Now, if this particular bismuth compound is hard to find where you live, don't despair! A decent alternative is tin(II) 2-ethylhexanoate, also known as stannous octanoate. This substance is commonly sold as an accelerator for condensation-cure silicones; for example, a nearly pure form is available under brand names such as Smooth-On Accel-T, Quantum Silicones QSil STO, or Bluestar VICURE #2. Just be careful not to buy anything based on dibutyltin dilaurate, dimethyltin dineodecanoate, or a similar tin(IV) compound: they will work great, but also happen to be a lot more toxic.

As with bismuth, the appropriate dosage varies depending on the resin; for OC-7086, a 1% solution, added at 1-2 drops per 10 ml of isocyanate, is a good starting point. Note that the compound is a bit more harmful than bismuth - handle it with care.

Bismuth and tin aside, there are several other, more exotic options to choose from. They may offer very specific benefits, such as improved curing characteristics in particular systems, or no subsequent inhibition of platinum silicones (which are somewhat sensitive both to bismuth and tin). If you need additional guidance, [click here](#) to expand a section with some rough notes.

**Random rant:** as noted above, most suppliers of lab chemicals are no longer willing to ship to residential addresses. Such restrictions make some sense for haz-mat materials - but the companies simply won't do **any business with you, even if all you're trying to buy is salt or glucose.**

There are two reasons for this. First, there is a growing number of government agencies - ranging from DHS, to DEA, to CPSC ([yes, that's right!](#)) - that don't want people to make anything ranging from illicit drugs to bootleg fireworks. Companies that sell to individuals face a hodgepodge of regulations and vague reporting requirements, and risk police raids and other serious consequences if they mess up. Second, there are liability concerns: if a kid loses an eye and his parents sue - well, even if the manufacturer prevails in court, there are still legal expenses and bad PR to deal with.

Because of this, it simply makes no sense for most of them to cater to the hobbyist market at all - shipping to a commercial address creates a pretense of due diligence, no matter how weak it may be.

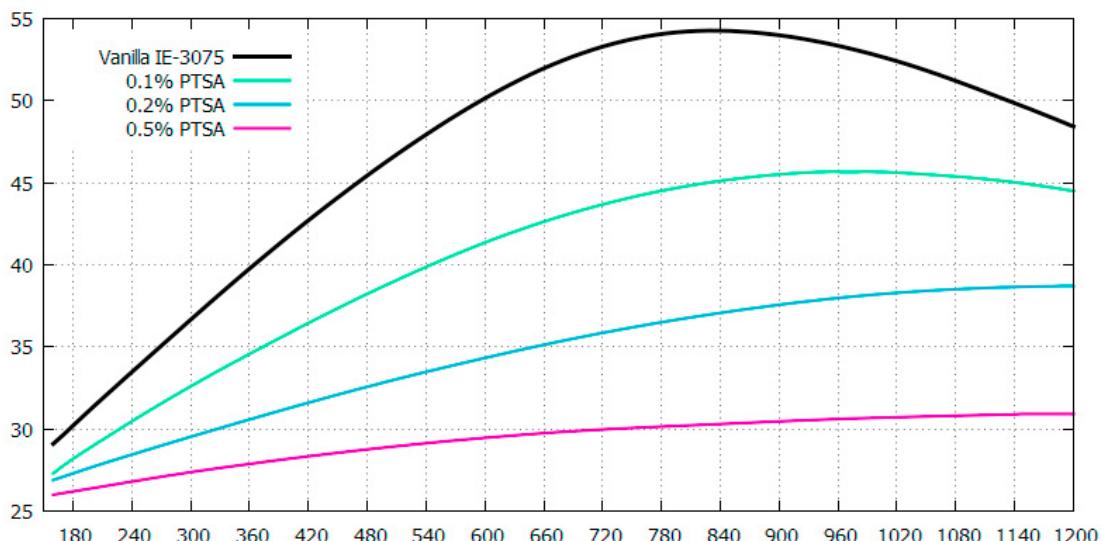
*To keep chemistry alive as a hobby, I urge you to support the remaining few places that did not succumb to this trend; in particular, consider going with eBay sellers or friendly outlets such as Chemsavers even if you have an opportunity to order certain chemicals directly from the manufacturer for less. Just stay away from [ScienceLab.com](#).*

#### 4.4.4.2. Slowing down the reaction

In many types of polyurethane formulations, it is possible to slow down the reaction by converting the catalyst to a less reactive complex. In particular, systems that rely on amine catalysts (and do not contain reactive amines as crosslinkers or any other vital components of the formulation) can be slowed down with strong, non-oxidizing acids that react with the catalyst to form a largely inactive ammonium salt. In the same spirit, some of the less obnoxious thiols and certain other substances can chelate a variety of organometallic catalysts.

In products such as IE-3075 or OC-7086, you can get good results with p-toluenesulfonic or methanesulfonic acid, both of which are available from Chemsavers for around \$20. Methanesulfonic acid is slightly more convenient, because it is liquid at room temperature; but p-toluenesulfonic acid is pretty easy to directly dissolve in polyol. Levels around 0.1-0.5% by weight should have a very pronounced effect; just be careful not to go overboard: excess acid may react with isocyanates and mess things up.

This graph shows the impact of p-toluenesulfonic acid on the curing exotherm of IE-3075, using the setup outlined earlier on:



If PTSA or MSA are not easily available in your market, a much less potent but possibly still acceptable alternative is sulfamic acid, a common cleaning compound available on [eBay](#) and [Amazon](#) for just a couple of bucks. The main problem with this compound is its relatively poor solubility in polyols and in most organic solvents. A saturated solution in n-methyl-2-pyrrolidone (NMP) may be your best bet; it will need to be added at a level closer to 1-2% by weight, which isn't exactly ideal. The solution is also not stable in the long haul, so prepare only as much as you intend to use in a couple of days.

What else? If you really can't get any of the above, you can try tartaric acid. Along with several other weak, aliphatic hydroxy acids, this compound will inhibit the reaction to some extent, although it shows some interest in side reactions that may liberate bubbles of carbon dioxide or impart a yellowish hue to your parts. On the upside, it's a common food additive, available pretty much everywhere; and it can be easily dissolved in acetone.

Note that PTSA, MSA, and sulfamic acid are all highly corrosive; use gloves and eye protection whenever working with concentrated solutions.

Oh, one more thing: keep in mind that while adding catalysts to a resin is guaranteed to make a positive difference, adding a particular inhibitor is not. Of the fast-curing resins discussed in this guide, I never found a way to significantly slow down TP-405x, but almost everything else seems to be a fair game.

#### 4.4.5. Blending several resins together

---

Every now and then, you may be hoping to modify the properties of a resin in a way that goes beyond what's possible with non-reactive fillers, plasticizers, and so on. Other times, you may be interested in changing its cure speed in a situation where the methods outlined in the previous section are impractical or simply don't work.

Well, the good news is that you can do quite a bit without resorting to making your own formulations from scratch. First of all, if you have two resins with comparably reactive isocyanates or polyols, and similar catalysts, you may be able to simply mix them together as-is. For example, let's say that you own HP-2150A and HP-2160D, and want to create a range of tough elastomers. The mix ratio is 100:43 for the first resin, 100:20 for the second one, and you want to blend them at a ratio of roughly 2:1 to get a rubber around 70 Shore A. In this case, suitable mixing amounts may be:

- HP-2150A isocyanate: 10 ml
- HP-2160D isocyanate: 5 ml
- HP-2150A polyol: 4.3 ml ( $10 \text{ ml} * 43 / 100$ )
- HP-2160D polyol: 1 ml ( $5 \text{ ml} * 20 / 100$ )

Of course, if the systems are based on dissimilar chemistry, the resin may cure prematurely, not cure at all, or have disappointing mechanical properties. Even in this case, not all is lost: you may be able to get somewhere by starting with a single resin, and then partly or completely substituting one of its components with that belonging to another system. There are situations where it won't work, and situations where it will give you useful materials with faster or slower cure profiles, and mechanical properties somewhere between these of its constituents.

The challenge with this second approach is figuring out the correct mixing ratio for isocyanate coming from product A, and polyol coming from product B; the manufacturer won't tell you how many reactive NCO and OH groups are there in each of the components, and without this information, you have to resort to trial and error. The correct ratio is usually between 100:30 and 80:100, and you can pinpoint it by doing several tests and selecting the range that resulted in the highest indentation hardness (Shore D durometer costs about \$25-\$50 on [eBay](#)); guessing the ratio within 5% should be fine in most uses.

#### 4.4.6. Heat-accelerated cure

---

As noted earlier, polyurethanes and silicones begin polymerizing the moment you mix the components; by the time you reach the demold time, the reaction is mostly over - but some cross-linking may continue for many days or weeks at an exponentially decaying rate. As this process goes on, the properties of the part will keep approaching these advertised in the datasheet.

If you are impatient and want to demold your castings sooner than normally possible, but don't want to sacrifice pot life or deal with chemical catalysts, placing the mold in a temperature-controlled oven will typically cut the time in half per every 10° C over ambient. Alas, the combination of significant thermal expansion of silicones (0.025%/°C), and the somewhat lower but still noticeable expansion of rigid polyurethanes (0.005%/°C), will probably affect dimensional accuracy of the part - so if you are aiming for snap fits, it makes sense to keep the mold at room temperature for as long as you can, and then bake it at no more than perhaps 40° C.

For an already demolded part, post-curing is a valuable process that involves fewer trade-offs, and lets you reach the final properties of the material in hours, rather than weeks; since the resin is already largely polymerized, and is not confined in an expanding mold, its own thermal expansion is less likely to have lasting effects. It's important to ramp up the temperature gradually, though, so that the part doesn't get too soft. I suggest one hour at 40° C, followed by 30 minutes at 60° C and 80° C; the cycle can be wrapped up with 1-3 hours at 100-110° C. Note that many polyurethanes begin to deteriorate around 150° C, and that for transparent formulations and flexible rubbers, this limit may be even lower.

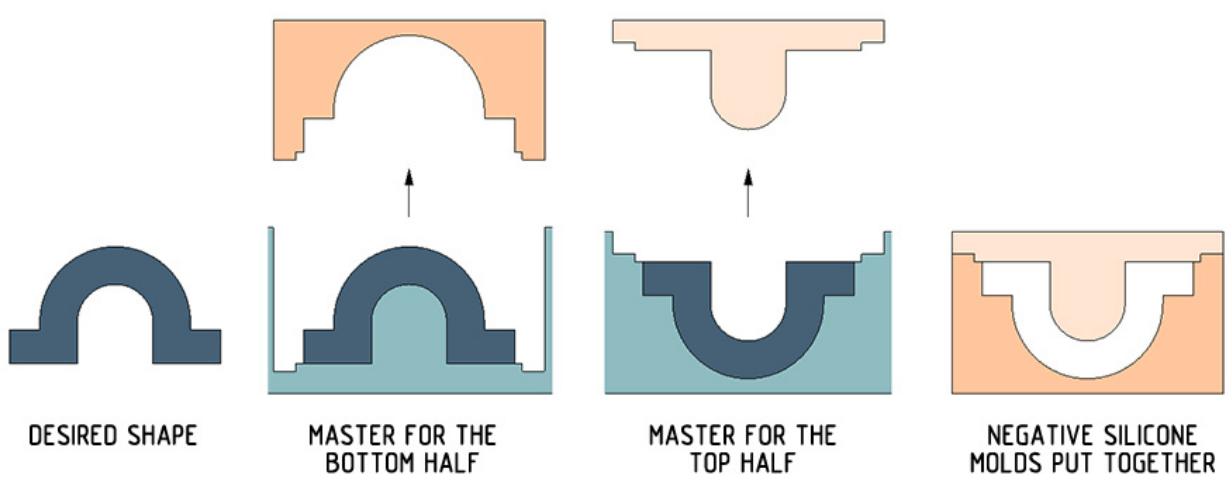
#### 4.4.7. Multi-part molds

---

All right, all right - enough with chemistry. But there's one more topic that may help you in casting work. Sooner or later, you will need to make parts with complex features on multiple sides. When replicating hand-made shapes, the

process is usually very intuitive; for example, the geometry can be submerged in a blob of silicone that is carefully dissected with a scalpel or a box cutter, and put together later on.

The process for designing accurately meshing multi-part molds in CAD software isn't much more complicated, but may take some effort to wrap your head around it. Let's say you want to make a part with a cross-section as shown on the left (and some additional features that prevent us from simply laying the shape on its side):



The first step is to make a regular mold similar to what we would do for one-sided parts, but also add a small pedestal around the geometry - this will serve as a registration mark. The second mold is simply the same part and its pedestal, flipped around; this top mold will neatly slide into the bottom one. Voila!

In more complex molds, the parting line may be located less conveniently, and may not allow all the air to escape on its own. In these cases, the mold will typically have a sprue through which the resin is poured in, and strategically placed vents to allow the air to escape from tight spots; a reservoir of resin on top of the sprue will offset for shrinkage in large parts, too. All in all, it's not hard, but when it comes to that, you will need to practice a bit.

## 5. Essential parts for robot builders

Righty-o. Now that we have the basics of machining, moldmaking, and resin casting covered to a good extent, it's useful to discuss more practical designs and part geometries of interest to robotics and other mechanical work. But before we dive into the inner workings of custom-designed gears and drivetrains, it's important to take a brief detour, and go over some of the prefabricated components that may come handy in your projects. Without a good source for tiny screws, springs, dowel pins, or cheap electronic sensors, you simply won't be able to get far.

Rest assured, this chapter will merely scratch the surface of it all. If you are looking for inspiration, get "[Machine Devices and Components Illustrated Sourcebook](#)" by Parmley, or "[Mechanisms and Mechanical Devices Sourcebook](#)" by Sclater and Chironis; and if you need a primer on electronics, my [concise guide to electronics for geeks](#) may come handy, too.

### 5.1. Mechanical components

#### 5.1.1. Screws and threaded rods

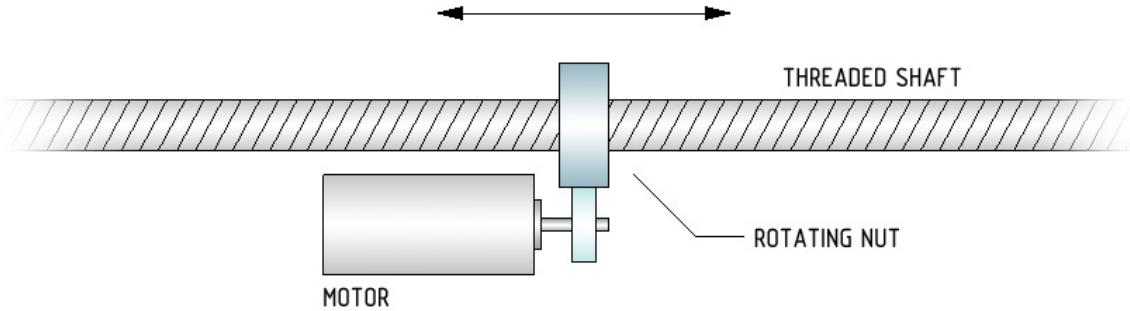
Machine screws, also known as bolts, are one of the most important items to put on your shopping list: if you want to create durable, serviceable designs that can't be approximated with indiscriminate use of glue, suitable fasteners are simply a must. In tough plastics such as polyurethanes, the use of screws is particularly easy: simply machine a slightly undersized hole and drive the fastener into it, impressing its own thread onto the part. Much of the time, you don't need a nut!

If you want to shop for useful sizes and quantities of machine screws, skip your local hardware store. There are three particularly good online sources for miniature and subminiature fasteners in bulk: [Micro Fasteners](#), [Fast Metal](#)

[Products](#), and [Amazon Supply](#). Micro Fasteners is a good all-around source for low-cost screws in diameters over 1.5 mm or so; FMP offers decent pricing on fasteners smaller than that. Last but not least, Amazon Supply (formerly Small Parts) tends to be a tad more expensive - but Amazon Prime customers get free two-day shipping on every single nut and bolt, so especially for small orders, it's quite a good deal.

The exact selection of fasteners depends on the projects you intend to pursue, but I recommend starting with a good assortment of 0.8 mm, 1.5 mm, and 2 mm screws (000-120, 0-80, and 2-56 designations in ANSI UTS, respectively), 100 pieces each. You should grab the cheapest variety of steel or brass screws, aiming for lengths around 4, 6, 8, and 15 mm; drive type doesn't matter a lot - could be slotted, Phillips, or hex. Expect to pay around \$2-\$5 per 100 pieces for common diameters, and closer to \$10-\$15 for sizes under 1 mm. Getting some nuts and washers, especially for 2 mm screws, is not a bad plan - but as noted, you won't be routinely needing them.

You may also want to look into threaded rods, available from sources such as Amazon - the diameter around 2 mm probably being the most useful. Their more boring use is an extended-reach screw (with one nut at each end); a more interesting possibility is creating extremely compact and simple linear motion systems, like so:



Another possible arrangement is using a motor to directly rotate the shaft. In both cases, the transmission enjoys a very high ratio, because every turn of the motor moves the nut by a distance equal to the pitch of the shaft - often in the vicinity of 0.5 mm or so. The downside is poor efficiency - likely under 20% - due to significant friction under load.

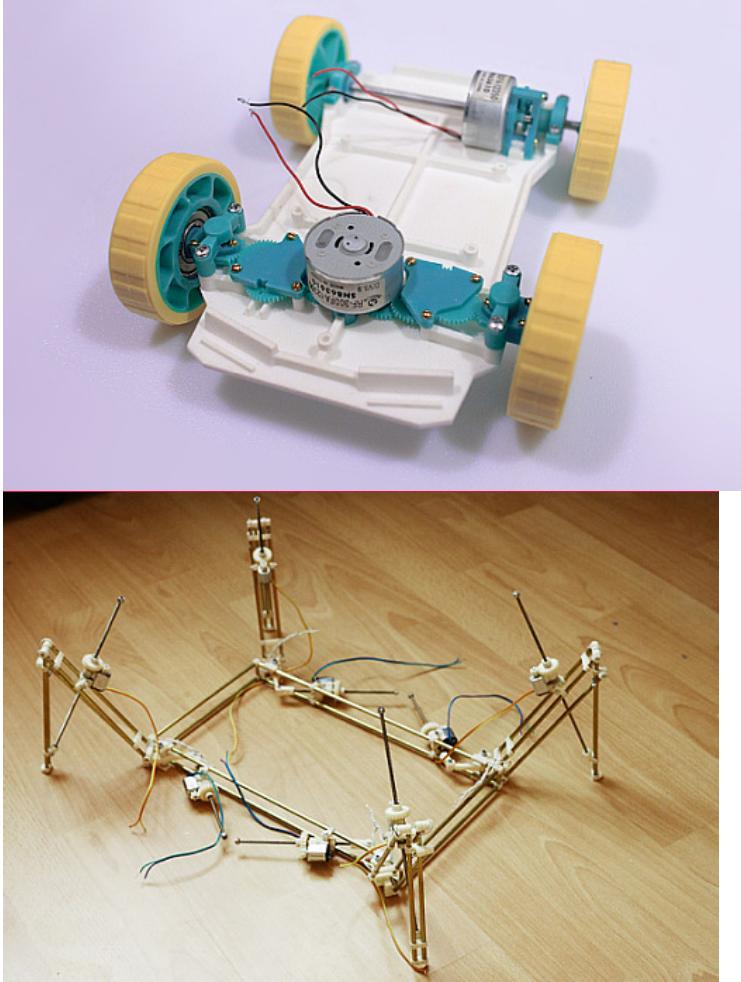
**Note:** when it comes to online retailers, many hobbyists also love [McMaster-Carr](#) as a source for screws and other mechanical components. That said, they are almost always significantly more expensive than specialized distributors, and often more expensive than Amazon. They do have good deals on some otherwise hard-to-find stuff - say, miniature timing belts - but it's more of an exception than a rule.

### 5.1.2. Dowel pins, rods, and tubes

Traditional dowel pins are rather unassuming: they are just pieces of featureless, cylindrical steel, machined to tight tolerances. For their appearance, they find a truly surprising number of uses: as axles for spur gears and other rotating parts; as registration pins for molds and multi-part assemblies; as movement limiters and contact sensors; as serviceable torque couplers; and so on. You just need to have some - trust me on that.

Non-tapered, solid metal dowel pins are available from many sources, including Small Parts / Amazon Supply, and cost very little - usually in the vicinity of \$4 to \$8 per 100 pieces. I suggest stocking up on 2 mm diameter pins in several lengths ranging from 4 to 20 mm. For high-precision work, 1 and 1.5 mm diameters may come handy, too.

Dowel pins aside, it's also good to have some vanilla steel rods or tubes: they are very cheap (usually \$1-\$2 per meter), and can be cut to size with a hand saw to build anything from long-reaching axles (left) to fairly complex frames (right, also showing threaded rods used as linear motion systems):



Metal bars with rectangular, hexagonal, or L-, I-, or T-shaped cross-sections are particularly useful for torque transfer, because you can simply slide components onto them, and there is no risk of slippage under radial load; perfectly round profiles may require the application of glue or the use of a lock screw.

### 5.1.3. Springs and spring wire

Similarly to dowel pins, springs have quite a few uses; many of them are obvious (wheel suspension and other pre-tensioned mechanisms, energy storage, etc), but some aren't. For example, springs are indispensable for transferring [rotary motion at an angle](#) - a process that otherwise requires complex bevel gears or [universal joints](#).

Perhaps the most common sort is a compression spring: it has generous spacing between its coils, and is meant to contract under load. You can find them inside many types of pens, spray bottles, and so on. The other popular type is an extension spring: it is tightly wound, and offers little or no compressive action - but stretches very well.

It's difficult to recommend a particular selection of compression and extension springs up front, but it's definitely a good idea to have a robust variety always available in your workshop, simply to prototype stuff easily. Possibly the best and least expensive assortment I have seen so far is [this set](#) - 200 reasonably sized springs for less than \$9; [comparable kits](#) are also available on Amazon. A great selection of individual springs with specific diameter, pitch, and length, can be also found on Amazon, usually in packs of 10.

Traditional springs aside, you should also grab some spring wire (also known as music wire). It comes handy for making contact sensors (especially whiskers!), for creating simple tensioners, and for designing other devices where you want to use a straight piece of elastic material to deflect effortlessly, and then spring back to its original shape. There are many low-cost assortments you can find on the Internet - and as usual, [Amazon isn't bad](#).

### 5.1.4. Ball bearings

There are many situations where it is desirable to constrain rotary movement to a particular axis of rotation, and

support it so that the part doesn't wiggle back and forth, or snap under load. Sleeve bearings are the simplest solution: you can route the rotating part through a round, slightly oversized opening, and perhaps use a bit of grease to minimize friction.

Alas, this approach has its limits: if the part is rotating very quickly, or if it's subject to significant radial forces, sleeve bearings will result in significant power losses or excessive wear. In particular, sleeve bearings for propellers and wheels may have a very limited lifespan.

Because of this, you should get a decent assortment of [ball bearings](#), and use them when appropriate. There are many sources of bearings on the Internet, but most of them tend to be pricey; [VXB.com](#) is a notable exception to this rule. They ship internationally and have an amazing selection of 10-, 20-, 30-, or even 100-packs at sensible prices - often hovering around \$1 to \$1.50 per piece. Some comparably good or even better deals can be found in the [\\$0.99 discount bin](#) or the [10-pack-section](#) of Boca Bearings, too - although their "regular" prices are higher than VXB.

Some of my favorite bearing sizes (ID x OD x H) are: 3x6x2 mm ([link](#), \$1 a piece, only for miniature projects); 6x10x3 mm ([link](#), \$1.50); 8x12x3.5 mm ([link](#), \$1.50); and 8x16x5 mm ([link](#), \$1). For larger projects, 8x22x7 mm bearings are a bargain, too - trading for about 50 cents a piece or less ([link](#)).

If you don't have any specific designs in mind, but plan to work on small to medium-scale projects, grabbing a set of 6x10x3 mm or 8x12x3.5 mm bearings is not a waste of money.

### 5.1.5. Gears and timing belts

When building precision mechanical assemblies, it is usually much more convenient (and a lot cheaper) to make your own gears than to source them online. Part of the reason for this is that most of the time, you will want to use compound gears with very specific diameters, heights, tooth counts and ratios, axle bore sizes, and so on. Chances are, you won't be able to find just the right ready-made options online, at least not on the cheap.

Timing belts are a different story, in part because they are tend to be simpler. Whether you need timing belts is a good question; they do offer an alternative form of gearing in applications where other solutions are inconvenient to use because of size constraints or the distances involved. For example, they may offer a space-saving way to couple a motor to a wheel without reducing ground clearance. Another popular use is in linear motion systems, where they can acutate any assembly clamped to the belt (and mounted on some sort of a rigid slide to keep it from flopping around).

Historically, timing belts used to be fairly expensive and difficult to source in miniature sizes, but this has changed over the past few years. One of the best sources for small timing belts, many for around \$1.50 a piece, is [McMaster-Carr](#); check out the MXL series for some very good picks. Some of their smallest ones are just about 10 mm long and 3 mm wide.

Timing belts run quiet and have very little backlash, but they are less durable and less rigid than plastic gears; they may be somewhat less efficient, too.

## 5.2. Robot-related electronics

### 5.2.1. Motors

The selection of motors at your disposal is definitely the single most important factor affecting the ability to bring your electromechanical designs to life. It's also something very easy to get wrong - or get right, but grossly overspend on.

It is probably safe to assume that you are interested primarily in small, low-voltage DC motors; if so, the choice is roughly as follows:

- **Vanilla brushed motors:** the traditional variety, with rotating electromagnets and a mechanical commutator that powers them up in a particular sequence. The most important advantage of this class of motors is the broad availability of many different form factors, often at ridiculously low prices (\$1-\$2). That allows you to buy them by the dozen, and helps with rapid prototyping.

Unfortunately, these motors operate efficiently only at very high RPM, and deliver fairly low torque, requiring the use of external gearboxes in almost every application; such gearboxes aren't difficult to make, but they add some complexity. The motors are also electrically noisy, and their lifespan is limited by the wear of commutator brushes - an issue in continuous-duty applications.

- **Geared motors:** these typically consist of a small brushed motor coupled to an integral gearbox that reduces output RPM, and brings the torque up to a more useful range. They usually operate with decent efficiency (85-95%) and offer transmission ratios of 20:1 to 400:1 or so.

The main trade-off is that many of the lower-cost plastic gear transmissions are flimsy and rather unwieldy, so in many applications, it may be more desirable to create them yourself; higher quality alternatives, and planetary gearboxes in particular, often fetch as much as \$20-\$150.

As a rule of thumb, it makes little or no sense to buy anything that [looks like this](#); microminiature [metal gears](#) are a usually better investment, and if you look around, you may be able to find them under \$8 a pop.

- **R/C servos:** affordable motors in this category usually consist of a geared motor coupled to a gearbox, a potentiometer attached to the output shaft, and a small integrated circuit that uses this feedback loop to maintain a position requested by the external microcontroller. The huge advantage of servos is the ease of use: they eliminate the need for additional sensors, H-bridge drivers, and other widgets normally involved in the control of a regular motor.

Unfortunately, most of the low-cost servos are pretty bulky - a brick about 4-5 times the size of the actual motor within; most of them are also not capable of performing 360° turns, owing to the design of the internal potentiometer. On top of that, most of the plastic-gear servos are somewhat fragile - the sale of replacement gears is an industry of its own.

- **Vanilla brushless:** brushless motors deal away with mechanical commutation, and instead, have a rotor with permanent magnets, surrounded by stationary electromagnets that are switched electronically in a specific sequence. They are more durable than their brushed counterparts, and because of improved heat dissipation, occupy a smaller, more lightweight envelope.

The key challenge with brushless motors is that they require precise external control to operate with good efficiency, often incorporating current sensors or Hall effect detectors; this adds cost and complexity. The other issue is that almost all the commonly available brushless motors are designed for high-current, high-performance applications, such as model aircraft - and with their extreme speeds and power demands, they are not particularly well-tailored for projects with more modest needs.

- **Stepper motors:** steppers are similar to brushless motors, but use a different rotor design with multiple densely packed poles; this makes the motor turn only by a small, precisely defined degree every time one of the coils is switched (7.5° or 15° is common in small motors; 1.8° in larger ones). There is no position sensing; their high-resistance coils are designed to let them hold a particular position without overheating, but limit their ability to operate at high speeds (their efficiency drops rapidly past perhaps 300 RPM or so).

The primary advantage of stepper motors is their high precision, ease of operation, and competent performance at relatively low speeds, suitable for directly operating many types of mechanical assemblies. Their disadvantage is that they are essentially a constant-current device, sinking the same amount of power when freewheeling and under load.

- **Exotic actuators:** somewhat disappointingly, there aren't that many alternatives to plain old motors. Linear actuators are interesting, but typically prohibitively expensive (\$100 and more). Solenoids, often used in door locks, are pretty cheap, but usually of limited utility. Lastly, "muscle wire", a material that contracts slightly (5% or so) when heated up by the flow of electric current, may sound cool - but because of the small displacement and fragility, it's more of a novelty than a practical tool.

As you can see, there is no perfect solution. I personally prefer sticking to vanilla brushed motors and creating my own gearboxes, but if your patience can wear thin, servos or geared motors may be a better choice. For any of

these motors, you should definitely look at the following characteristics when shopping around:

- **Maximum rated voltage:** every motor is designed for a particular maximum voltage, past which it may overheat or blow up in some other fashion. You can operate any motor at as little as 20% of its rated voltage - but in such a case, you will be hauling around a lot of dead weight in the form of windings, magnets, and so on, that are grossly oversized and serve no useful purpose.

In general, motors rated for somewhere between 3 and 9V are probably of most use in small- to medium-size projects, owing to the availability of affordable batteries and power-switching transistors compatible with these voltages. Going beyond 12V can be expensive, and is best avoided unless you are building something big.

- **No-load RPM and stall torque:** these two parameters describe the behavior of the motor at two extremes: its speed when freewheeling with no load, and the turning force needed to effectively stop the motor in its tracks. In both of these cases, the motor is not doing any useful work, and its efficiency is zero. In between, the dependency between RPM and torque is roughly linear: with a load equal to 80% of the stall torque, the RPM will hover around 20%.

This dependency may suggest that instead of using complex gears, it may be possible to simply operate the motor with a carefully tuned load of 95%, thus reducing its speed down to 5%. Unfortunately, this is usually a bad idea. In essence, every motor has a specific efficiency curve - and for low-power DC motors, it typically looks [like this](#): the device is most efficient when the load is around 20%, and the performance drops off somewhat linearly past this point, eventually reaching 0% at stall. Therefore, in any application where power consumption and heat dissipation are of concern, it's good to operate the motor at somewhere between 5% and 50% of its maximum load most of the time; outside that range, you're just heating the air.

To understand what you will be getting out of a specific motor, and to compare different products, it is often useful to compute their normalized torque: the torque they will be able to deliver when paired with a gearbox that brings their speed to a particular value (say, 100 RPM). Assuming that you are aiming for a load of 50%, and that the gearbox has an efficiency of 85%, the formula will be:

$$\text{normalized\_torque} = (\text{stall\_torque} * 50\%) * (\text{idle\_RPM} * 50\%) / 100 * 85\% = \text{stall\_torque} * \text{idle\_RPM} / 470$$

- **Stall current:** this parameter should help you size the power supply and all the associated driver circuits. When multiplied by the supply voltage and compared to normalized torque, it may be helpful for comparing the efficiency of various motors, too.
- **Dimensions:** all things being equal, the smaller the better - but form factors matter, too. Long but slim motors are preferable for wheeled robots, as they result in optimum ground clearance when placed inline with or near the axle; while pancake-shaped motors with a large diameter but short length are much better when you need to cram them inside an articulated joint in a legged robot without anything sticking out.

Possibly the best source to find a great assortment of low-cost brushed motors (both vanilla and geared) is [Kysan Electronics](#); they have a \$100 minimum on all online orders, but seem to be willing to make exceptions if necessary. Good deals can be also sometimes found at various surplus outlets, including [All Electronics](#), [Electronic Goldmine](#), [Surplus Shed](#), [BG Micro](#), or [HSC Supply](#) - but their inventory can change rapidly, so your mileage may vary. Last but not least, for servos and brushless motors, [Hobby King](#) is hard to beat - they ship from Hong Kong, but do so promptly and cheaply; on orders under \$200 or so, you are unlikely to run into import duties.

Whatever you do, I'd recommend avoiding robotics-oriented sources such as [Solarbotics](#), [Robotics Connection](#), [Pololu](#), [Acroname](#), and many more. They are good people, but they usually sell exactly the same low-cost motors, and simply charge you more for the privilege of shopping with them. Case in point: [this motor](#) costs \$23 when bought from Robot Marketplace, or \$16 when you go to Solarbotics - but Kysan Electronics carries it for \$8 a piece... or just \$3 on orders over 1,000 (which is probably the price that the first two shops have paid).

In any case, it makes sense to find 2-4 models that are best suited for your needs, and then buy 10-20 pieces of each; having a steady supply of well-performing motors beats having one or two of every mediocre product

available on the market. My personal recommendations are:

▪ **Vanilla brushed:**

- [Mabuchi FF-N20PN \("8117" or "8721"\)](#) (\$1.00): one of the tiniest motors I found that can still pack a punch (pager motors are smaller, but offer negligible torque). Diameter of 10 / 12 mm, length of 15 mm. When operated at 3V, it idles around 13,000 RPM, and delivers about 15 g\*cm of stall torque - drawing 1.3A at that point. Great for miniature projects.

(That motor seems to be hard to find these days; FF-N20PA-11155 motors available from [Kysan](#) are a close match, but cost a bit more - just under \$2.)

- [Mabuchi RF-356CA-10250](#) (\$2.20): a reasonably powerful pancake-shaped motor for all-around use. Diameter of 24 mm, 8 mm long. About 11,000 RPM, 40 g\*cm, 600 mA at stall when running at 6V. [RF-300FA-12350](#) is a cheaper alternative, but it's 4 mm longer and has a lower speed.
- [Mabuchi FF-130SH-14230-6V](#) (\$1.90): an elongated all-purpose motor that delivers 13,000 RPM when idling, and stalls around 115 g\*cm, drawing 1.3A, when operated at 10V. Diameter of 15.5 / 20.5 mm, length of 25 mm. A comparable but slightly larger 12V alternative is [Canon EN22](#).

▪ **Spur geared motors:**

- [Sanyo 12GN-0348-NA3S](#) and [12GN-0348-NA4S](#) (\$7.50): two cheap, robust, extremely small metal gear motors delivering about 2,200 g\*cm (NA3S) and 3,500 g\*cm (NA4S) of stall torque, and idling around 100 and 70 RPM, respectively. Both run at 6V. Their tiny dimensions make it worthwhile: they measure just 12 x 10 x 25 mm.

▪ **Stepper motors:**

- [Symbol Technologies 21-02485](#) (\$1.00): about 20 mm wide and 12 mm thick; a 5-wire hybrid that is best operated in bipolar mode (where the polarity of the windings needs to be reversed, necessitating the use of H-bridge drivers), and runs fine at 9V. 15° per step, 100 mA per coil, around 40 g\*cm of torque. Unfortunately, appears to be out of stock right now.
- [Epson STP35NI48SV50](#) (\$3.00): a larger stepper motor, 35 mm wide and 12 mm thick. Unipolar operation (can be controlled with a couple of transistors), 7.5° per step, 12V, 250 mA per coil, around 220 g\*cm of torque.

▪ **Servos:**

- [Mini Man 4.3](#) (\$3.50): a really cheap, miniature 6V servo of acceptable quality, delivering around 500 g\*cm of stall torque. Its speed is about 10 RPM with no load.
- [Hextronik HXT900](#) (\$2.65): more powerful, slightly larger servo - 1,600 g\*cm at stall, 40 RPM under no load, runs off of 6V. Very popular with hobbyists.
- [Hextronik HXT5010](#) (\$4.50): a larger, ball bearing servo capable of delivering about 6,500 g\*cm at stall. It's slow - 8 RPM with no load at 6V. Reportedly can be modified to rotate 360°.

If you need inspiration, here's a video of Mabuchi FF-N20PN powering a miniature planetary gearbox:



### 5.2.2. Useful sensors

Sensors are essential in almost any electromechanical design, helping interact with the outside world, and providing internal feedback about the state of mechanical assemblies. This section covers some of the most useful, low-cost choices to consider in your work:

- **Tactile switches:** simple, small, cheap, and very versatile. When placed behind an elastic bumper, they can work as a wall sensor for wheeled robots; in conjunction with rubberized foot or a floating axle, they can sense contact of the robot body with the ground; and when used as a movement limiter, they are useful for detecting the "home" position of many assemblies.

A wide selection of microswitches in all shapes and sizes is available from [Mouser](#) and from other major electronics outlets. Pricing starts at \$0.10 or so a piece when bought in reasonable quantities; be sure to look at dimensions, trigger force, and button travel. In most cases, it makes sense to stock up on the smallest, low-profile non-SMD switches you can find - for example, Mountain Switch 101-TS4311T1601-EV (\$0.20). Larger ones - such as ALPS SKHCBFA010 (\$0.30) - may be useful when a greater pressure-sensitive area is desired.

- **Photointerruptors:** extremely useful for non-contact position sensing when you don't want to obstruct the motion of a component. They work by shining an IR LED on the element, and then measuring the reflected or transmitted light with a phototransistor. My favorite is a tiny, wide-slot Toshiba TLP841 (\$0.40); for reflective applications, Optek OPB609AX (\$0.55) may come handy, too.
- **Mechanical rotary encoders:** these can be used to measure the angular position of an assembly. They use mechanical coupling, and therefore add some friction - but often offer a convenient alternative to optical sensing. Linear sensors, such as ALPS RDC803001A (\$1.50), leverage a heavy-duty rotary potentiometer with several taps; pulsed devices, such as Mountain Switch 101-5433-EV (\$0.55), simply alternate between closed and open circuit at predefined angles, commonly producing 12 or 24 pulses per turn. Only linear sensors will provide the absolute position, but they cost more, and require an A/D converter to interpret the data.
- **Magnetic field sensors:** solid-state Hall effect sensors such as Honeywell S41 (\$0.50) can work very much like a photointerruptor, but will sense the position of a tiny permanent magnet embedded in the rotating element; their primary benefit is much lower power consumption, and the fact that they can "see" through opaque materials, enclosures, and so forth.
- **Distance sensors:** these can be positioned either in a fixed location on the robot body, or rotated to sweep the environment in a manner similar to a radar - creating a 2D image that is a lot more useful than the data that can be acquired with a digital camera, especially if you don't have too much computing power to spare. The two most common rangefinding options are infrared and ultrasound. Infrared sensors are cheaper, but have a limited sensing range, and may be considerably less effective in very bright light (e.g., outdoors); Sharp GP2Y0A02YK is probably the most interesting option here, selling for under \$15. For ultrasound, you must be prepared to pay a bit more; for

example, [MaxBotix LV MaxSonar EZ2](#) fetches around \$25.

- **Digital compasses:** these may be useful in robots that need to maintain precise registration with the environment over extended periods of time. You can choose from specialized solid-state Hall effect sensor ICs, such as Honeywell HMC1051; and regular Hall effect sensors that monitor the position of a traditional compass needle (an example of that is [Dinsmore](#) 1490 or 1655). Expect to pay \$10 to \$40.
- **Other fancy sensors:** some designs - especially flying robots - may benefit from accelerometers, such as MMA8453QT (\$1.40); flex sensors; heat sensors; inclinometers; tilt switches; microphones; digital cameras; or GPS units. That said, these are usually selected for a specific application, and there is no need to keep a continuous stock.

### 5.2.3. IC glue and MCUs

If you have a favorite brand of microcontrollers, there is probably no need to revisit this topic; but if you are looking for advice, it's pretty hard to go wrong with AVR chips such as [ATmega1284P](#) (\$8). This particular 8-bit MCU, for example, operates at speeds up to 20 MHz (internal oscillator is provided), has 128 kB of Flash memory for program storage, 16 kB of data memory (SRAM), and 4 kB of non-volatile EEPROM. It's essentially a complete computer-on-chip, complete with 32 bidirectional I/O lines, 8-channel 10-bit ADC, hardware PWM channels - all that supporting a wide range of supply voltages, from 1.8 to 5.5 V; there are precious few external components required to operate it in most real-world applications. ATmega chips have a mature GCC-based toolchain with tons of useful libraries, a nice emulator, and a pretty good IDE - and unless you are doing complex image processing or working on something else data-intensive, they will serve you well. (In more demanding tasks, you may need to spend quite a bit more on 32-bit ARM or AT32 chips; I also like [Intel Edison](#).)

For ATmega, the only other gadget you need is a simple USB ISP dongle (e.g., AVRISP mkII), costing somewhere between \$15 and \$30 - and even that can be avoided if you opt for a chip with a built-in USB controller.

*Note: some people love AVR-based development platforms such as Arduino or Teensy. I am personally wary of these boards, because I find them to offer very few real benefits over the AVR chip itself; you are essentially charged a 1000% markup in exchange for someone soldering the chip to a PCB, and then adding several components that are completely unnecessary in many uses, but make it look sophisticated (e.g., voltage regulators, external crystals).*

Especially when developing more complex software, you may find it useful to add a way for the MCU to communicate essential information in an easily readable way. Tethering it to a computer is one option, but you may also consider getting an LCD module based on a well-known HD44780 chip; for example, [NHD-0216K1Z-NSB-FBW-L](#) (\$11) is a very user-friendly device with ample display space. It can be controlled with as few as 6 data lines, and is pretty trivial to interface with - its dedicated controller maintains its own display memory, and even stores editable font data and track of the cursor for you, so you just have to send ASCII data to the appropriate port.

In addition to the MCU itself, you should also have a good assortment of standard "glue" chips that are useful for example in multiplexing and demultiplexing applications, and will allow you to extend the I/O capabilities of your chip almost arbitrarily. Probably the best IC family to stick to is 74HC - they are widely available and fairly cheap (\$0.10 - \$0.35 per chip), and offer respectable speeds and good load driving capabilities. You may want to grab basic logic gates (74HC00, 02, 04, 08, 32, 86 - NAND, NOR, NOT, AND, OR, and XOR respectively); line drivers (74HC240, 241, or 244); multiplexers, demultiplexers (74HC164, 165); line selectors (74HC137, 42, 151); and flip-flops / latches (74HC175, 75, 259). Some projects may also have uses for counters, timers (e.g., 7555), external oscillators, assorted op-amps, etc.

About the only thing you can't do with all these parts is driving any power-hungry loads, such as motors: the tiny transistors inside most MCUs and 7400 series chips can output at best around 20-40 mA per line - enough for a LED or two, but not much more. It is possible to use discrete power MOSFETs (e.g., BUK7510) to control high-current devices, but doing so is not always space- and cost-efficient - so you may want to look into IC-based motor drivers. FAN8082 is probably the cheapest (\$0.40) full H bridge (i.e., bidirectional) driver capable of delivering up to 1.5A at 18V to brushed motors and bipolar steppers; it even comes with rudimentary speed control. The disadvantages of this chip are its relatively high voltage drop (almost 2V), and the fact that it doesn't support

"freewheeling" (high impedance) mode. Somewhat more expensive TA7291P (\$0.90, 2A peak at 20V) supports all four output states: forward, reverse, brake, and freewheel; TLE52052 chip (\$3.50, 6A peak at 40V) can drive even larger motors with ease. Several dozen similar products exist - shop around, and grab at least around 10 pieces or so.

For driving unipolar steppers, solenoids, and other power equipment where you don't need to change polarity, you can also save some money by going with simpler devices: ULN2003 (\$0.30) can drive up to 6 devices at 500 mA and 50V (or one device at 3A); while ULN2065 (\$2) has four outputs capable of delivering 1.5A at 35V, adding up to 6A total.

## 5.2.4. Power sources, watchdogs, regulators

About the last major set of electronic components that you need to think about are the power sources you will be using in your work. To make the right call, you need to consider several factors:

- **Nominal voltage:** in most cases, you should use a power source that provides perhaps 10-15% more than the voltage required by the most power-hungry components of your design - and that's usually the motors. In general, somewhere between 3.5 and 11 V is ideal.
- **Capacity:** try to estimate the power needs of your circuit, and aim for something that would give you a sensible run time on a single charge. For example, if you are expecting to sink 100 mA on average, you will get up to 8 hours out of any 800 mAh cell.
- **Peak current / ESR:** make sure that the cell can accommodate the expected peak current draw of your device, with a generous safety margin. This matters: for example, a typical 9V battery will typically source no more than 1A - and once you exceed that, the voltage will drop dramatically.
- **Weight and dimensions:** there are substantial differences in power density between various cell types, so pay close attention to these characteristics, especially if you are trying to build something lightweight.

Today, the best all-around option for robotics are rechargeable lithium-polymer cells, simply because of an excellent balance between capacity, weight, and cost. My favorite source is [Hobby King](#). They have good products, and although they are in Hong Kong, they ship cheaply, quickly, and with no hassle whatsoever. If you browse their site, you can find a 7.4V 5 Ah cell, weighing around 300 g, for about \$25; a smaller 1.6 Ah cell fetches \$10 and tips the scales at 90 gram; while a tiny 800 mAh one weighs barely 50 g and costs \$5.

Of course, nothing comes free: lithium batteries have two drawbacks that you should know about. First of all, if they are charged improperly or badly damaged, they can overheat and catch fire - so you need to store and handle them with some care. The other issue is that they shouldn't be discharged past a certain minimum voltage to avoid altering their chemistry; using a voltage cut-off IC, such as MAX8211 or MAX8212, is a very good idea.

Of course, there are many alternatives to Li-poly; a typical AA battery is nothing to sneeze at, and delivers up to 3 Ah, with peak current as high as 10A; your usual 9V battery is closer to 500 mAh and can't source more than 1A. If weight is not an issue, you can also go with lead-acid batteries, of course: they are cheap, but weight a ton (\$20 will get you 15 Ah at 6V, but be prepared to haul around 2 kg). Ultracapacitors are also of some interest in recent years - but right now, they tend to be fairly expensive, especially if you are interested in supply voltages over 2.5V or so. Last but not least, solar cells deserve a honorable mention - although similarly to ultracapacitors, they are not that practical in everyday uses. Because of their lamentable power capabilities in function of their size, they are useful mostly as a way to conveniently recharge a chemical battery or a capacitor, and not as a continuous primary supply.

Oh, one more thing: for prototyping, I recommend grabbing an adjustable benchtop power supply, such as [Mastech GPS-3030D](#) (\$90); convenience of being able to quickly adjust voltage aside, their huge benefit is that you can limit the current to a safe value, so that an accidental short-circuit will not destroy everything in its path. The same can't be said about most batteries.

**Tip:** the reason why you should match the supply voltage with the most power-hungry components in your circuit is that high-current DC voltage adjustments can be pretty tricky. It's easy to lower the voltage

*supplied to a low-current device, such as a microcontroller or a couple of LEDs: just grab a linear regulator such as LM317T (\$0.25) or L7805 (\$0.50) and be done with it. For efficient regulation of higher currents, or for stepping the voltage up, you generally need switched regulators, however.*

*Such regulators are fairly complicated to build on your own, and get expensive if you want a plug-and-play solution. For example, ICL7660 - a chip that can handle up to 20 mA - goes for \$2; Murata OKR-T3-W12-C - a hybrid device that can deliver up to 3A - retails for \$7; and a 6A variant of the same Murata product will fetch \$14.*

### 5.2.5. Other components

Well, it goes without saying that you will also need an assortment of generic electronic components to get anywhere: make sure that you have a bunch of resistors and capacitors, a handful of PCB mount potentiometers, some medium-power MOSFET transistors (n- and p-channel), a good selection of terminal blocks and ribbon connectors, a solderless breadboard or two for prototyping, perforated boards in various sizes, and so on. In fact, if you need any help with selecting the right components and using them in a circuit, check out my [short primer on electronics](#) in your spare time.

What else? If you want to make your own PCBs for finished projects, you can of course print and etch them - although keep in mind that it's also quick and easy - and often more precise - to machine them on your CNC mill: you can simply selectively remove copper plating from a blank board with a cutter - and drill mounting holes at the same time.

And of course, don't forget about installing a conveniently located and obvious self-destruct switch!

## 6. Practical part geometries

To wrap up the discussion of technical topics, let's go over a collection of practical approaches that should help you create useful and reliable mechanical designs for small- to medium-sized projects - no matter if you are using CNC machining, 3D printing, or still carving stuff in stone. The failure to get even the most rudimentary understanding of these topics is one of the cardinal sins of hobbyist makers - so don't let this happen to you.

### 6.1. Basic part design tips

The mathematics of material engineering are fairly obtuse, and the finite element modeling software that can be used to model the dynamic behavior of real-world part geometries is prohibitively expensive. Nevertheless, if you are designing small- to medium-scale components and can afford to learn from mistakes every now and then, there is a couple of simple rules of thumb you can follow to get great results without resorting to any of that.

(You can also find quite a few good engineering tips on [this page](#).)

#### 6.1.1. Rigidity management

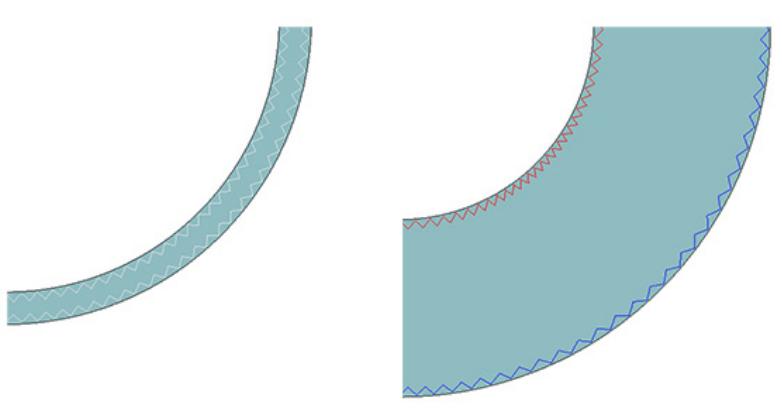
Perhaps the most profound cognitive dissonance you can experience during your first adventures in DIY manufacturing is making the first part that happens to be long, thin, and flat. Let's say you grabbed IE-3075 and cast a piece that is 2 mm thick, 1 cm wide, and 15 cm long; you will be probably impressed by how difficult it is to break this sample - but also shocked by how easily it deflects. It just doesn't seem to make any sense: the datasheet says that the material is ought to be 50% more rigid than the plastic used to make Lego bricks, and several times less flexible than the commodity plastics used in everything from toaster ovens to computer cases. But your toaster doesn't flex nearly as much, right?

To understand what's going on, consider what would happen if you applied the same bending force to the side of the part, rather than the top:



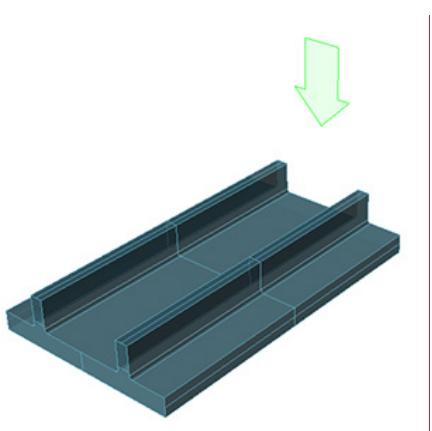
If you do that, you won't be able to flex the part even the slightest bit. The cross-section of the material is still the same - 2 x 10 mm - but all of sudden, it is a lot more willing to resist your efforts.

That's because flexing the material amounts to stretching the polymer chains farther apart on one side of the part, and squeezing them close together on the other end. If the material is thin, the effect is relatively modest, and you don't need to invest a lot of energy to make it happen. For thicker segments, the displacement is a lot more dramatic, however:



The effect is pretty significant. Simplifying slightly, the rigidity of the part is proportional to the width of the sample - but to the cube of its thickness.

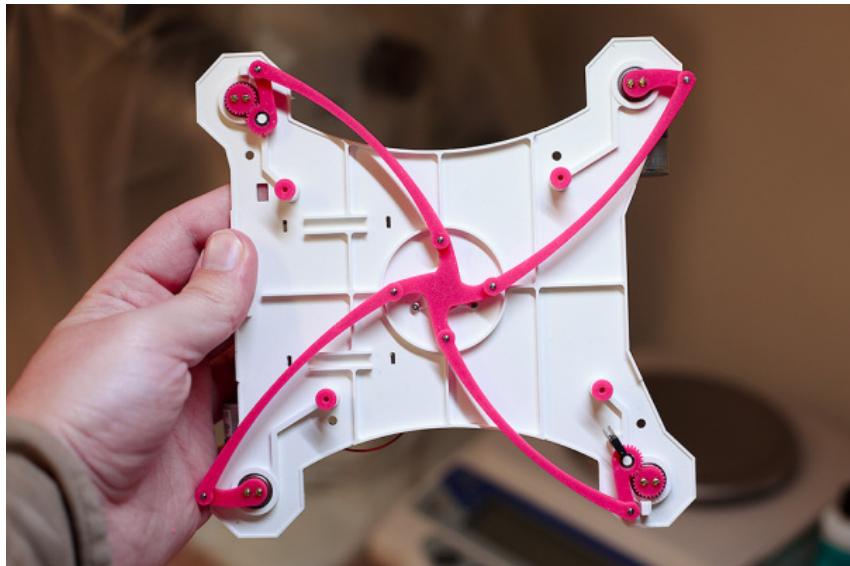
This behavior brings us to the manufacturing trick that is being used in almost every item made out of plastic or sheet metal - including toasters and Lego bricks. In essence, you can greatly improve rigidity by increasing the maximum span of the part in the direction subjected to bending loads - without adding any significant bulk. The most common approach is the addition of a couple properly oriented ribs:



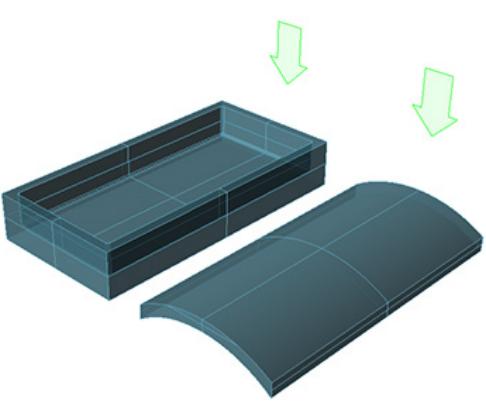
The average thickness of the part - and thus its weight or the amount of plastic used - hasn't changed significantly. But because each of these ribs contributes to rigidity in proportion to the cube of their height, their impact is more

significant than it may seem.

The following picture is a good example of the use of ribs to control the rigidity of a thin-walled chassis of a medium-size robot. Note how the ribs are routed around any locations where additional clearance is required, and not interfere with the mechanical design. In fact, they probably improve the overall aesthetics:



It is also possible to achieve the same effect without having to introduce any "standalone" ribs; for example, if the part uses curved surfaces or happens to have outer walls, this can provide a comparable degree of reinforcement:



Properly reinforced thin-walled parts weigh less, cost less, provide more clearance for other components, exhibit much less shrinkage - and often simply look better. Because of this, try to learn and use these approaches as often as possible. Stick to 1 to 1.5 mm as a baseline wall thickness for all the larger parts - especially for chassis components, covers, etc - and just sprinkle them liberally with ribs, gussets, and other features of this sort. A handful of ribs extending 2-4 mm from the surface of the part is usually all you need to make it work.

### 6.1.2. Stress distribution

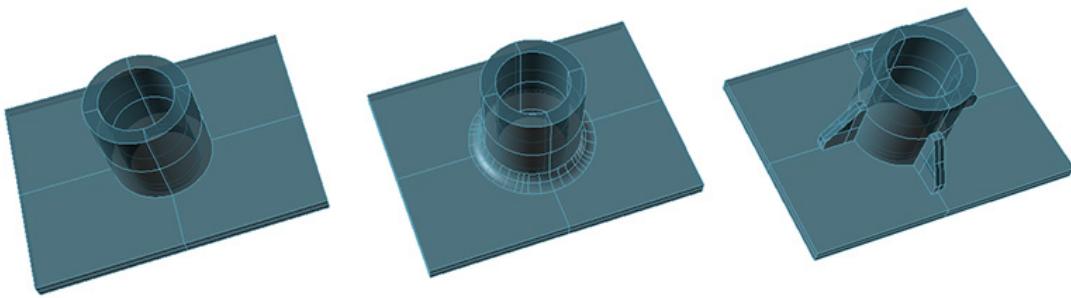
So far, we've talked about the need to reinforce thin, long, planar surfaces; but there is one more situation where a small amount of extra plastic routinely saves the day.

From the discussion of casting resins earlier in this guide, you may recall that the resistance of these materials to non-uniform loads (e.g., tearing) is dramatically lower than the ability to withstand uniformly distributed stress. For this reason, you should pay close attention to sharp corners, jagged edges, holes, and similar features in locations subjected to significant loads - and incorporate simple reinforcements to ensure that the stress isn't concentrated in any single spot.

For example, when designing a coupling for an axle that may be subject to strong radial forces and shock, adding a fillet or a couple of load-bearing gussets can save the day; all other things being equal, the "vanilla" approach

shown on the left is much more likely to fail:

---



Given the negligible cost of these features, it typically makes no sense to spend too much time trying to figure out in which exact spots they happen to be absolutely necessary. Whenever it is clear that a localized stress may occur, throwing in a rib, a gusset, or a fillet is simply a good plan.

### 6.1.3. Fitting tolerances

Another problem that a novice designer may bump into is the question of fitting tolerances: when one part needs to mate with another, what should their respective dimensions be to form the desired type of fit?

The correct answer depends on a number of factors, including the accuracy of your CNC mill, or the operating conditions of the assembly (temperature, risk of contamination, running speed). Nevertheless, when working with a reasonably accurate milling machine and dealing with normal circumstances, the following rules of thumb tend to work pretty well for rigid plastics:

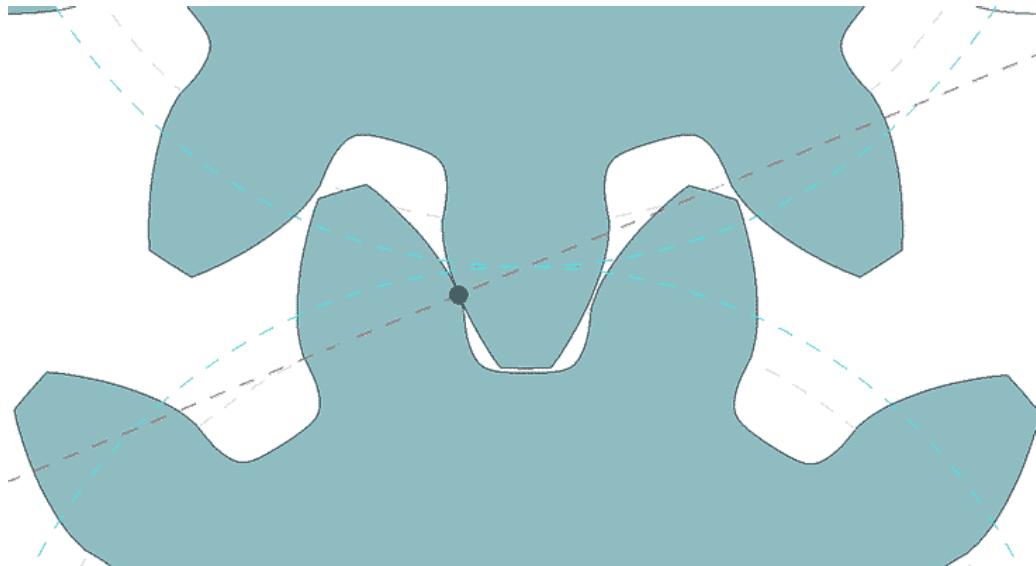
- **Snug fit:** if you want the parts to mesh tightly - for example, to be glued or fastened together - make the opening exactly the same size as the mating part.
- **Interference fit:** if you want the parts to hold together without any additional fastening, you can undersize the opening by about 0.01 to 0.02 mm (i.e., offset the profile curve by 0.005 to 0.01 mm). Note that you will need to use force to make them fit, so this may be not an option for fragile parts or unusually brittle resins.
- **Low-friction meshing:** to allow a round element to rotate with minimal friction, it's usually sufficient to leave a clearance of around 0.02 mm (offset of 0.01 mm). Note that for high radial loads, ball bearings may be preferable to sleeve couplings.
- **Self-tapping holes for machine screws:** measure, look up, or calculate the minor diameter of the screw (the outer diameter minus the depth of the thread). Use this as a diameter for the hole. A drop of WD-40 will help drive the screw in for the first time.

When in doubt, be quick to experiment: it takes just a couple minutes to machine a rectangular or cylindrical test piece and play with it. Save yourself the time and frustration of having to debug issues in more complex parts.

## 6.2. Creating spur gears

Gears are an extremely versatile tool that will come handy in all sorts of mechanical assemblies. Depending on your needs, they can reduce or increase the speed of movement, adjust torque, or change the direction, axis, and type of motion - all that with high precision, excellent efficiency (95%+), and great load-bearing capabilities. Alas, unfortunately for hobbyists, much of the freely available information about the geometry of common types of gears is misleading or incomplete - and even the articles posted on Wikipedia are littered with inaccurate images or animations such as [this](#).

The key property of a well-designed set of gears is that they transfer motion almost exclusively through a rolling action, with very little sliding (and thus friction) involved; and that they maintain constant velocity and a constant angle of contact through the process. If you violate these principles, your gears will become inefficient, noisy, and prone to vibration. The optimal design is exemplified by the ubiquitous involute spur gear:

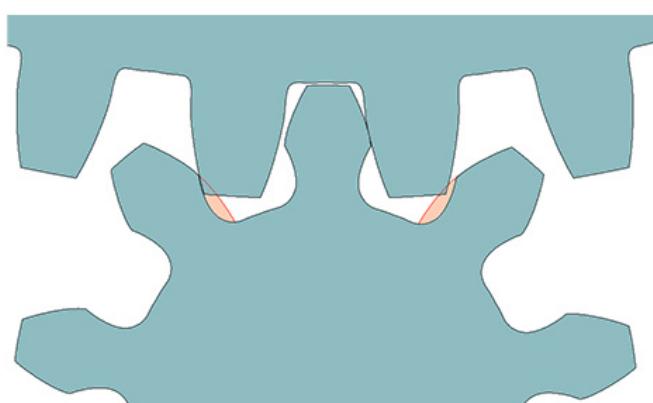


The blue lines shown on this picture are the theoretical pitch circles of the gears; if you replaced the gears with idealized rubber rollers of these diameters, the behavior of the transmission would not change. The linear speed of any mating gears is the same when measured along the pitch circle, and the ratio of angular speeds depends on the ratio of their pitch diameters. For example, if one of the gears has a pitch circle with a diameter of 10 cm, and it's turning another gear with a pitch diameter of 4 cm, one turn of the former will rotate the latter by  $10 / 4 = 2.5$  turns.

Before getting any real work done, you need to make up your mind about several key parameters that will define the gears used in a particular project; except for tooth count, these values should be the same for any two gears that are supposed to mesh with each other.

### 6.2.1. Tooth count

This parameter depends chiefly on the transmission ratios you are hoping to achieve. The only constrain here is that with very low tooth counts, you will see a behavior known as undercutting: the two meshing gears will collide, necessitating the introduction of some additional clearance - and thus weakening the part:



This is not a big deal if the undercut is small; but when it begins to extend significantly into the area where power transfer takes place - that is, the region above the so-called base line, the gray circle shown in the earlier animation - the performance of the gear may be compromised. To avoid having to heavily compensate for this issue, unless exceptional circumstances apply, try to keep tooth counts over 8-9 or so.

**Note:** most of the gear generators you can find online can't deal with undercuts at all: quite simply, if you try to specify fewer than a certain number of teeth, they will reject your input or generate an incorrect result. Avoid these like a plague - and scroll down for better alternatives.

As for the maximum number of teeth - there is no real limit to how far you can go, but to keep your models simple and the cutting process rapid, it's preferable to keep the count under 100 or so.

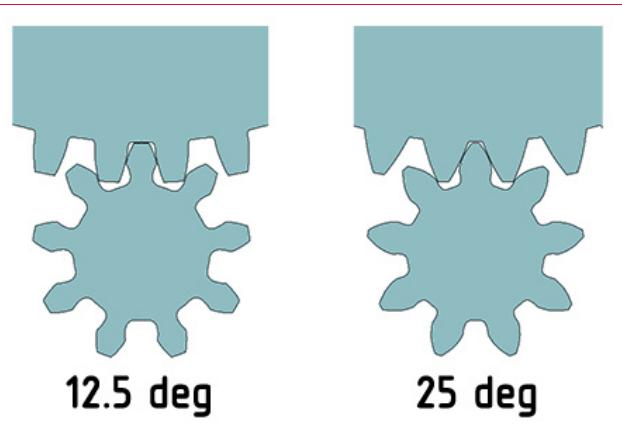
### 6.2.2. Tooth width

This parameter is measured at the pitch circle. The tooth must be wide enough to allow the mating region to be machined with the tools you happen to have at your disposal; in fact, it's advisable to use tooth width at least 30% greater than the diameter of the smallest end mill in your collection.

In precision applications that do not involve extreme torques, the value will typically range from 0.3 to 1 mm - and 0.6 mm is a good starting point. You may want to increase the width when working with larger models, to keep the number of teeth sane.

### 6.2.3. Pressure angle

When you look back at the animation shown earlier in this section, you will notice a slanted line that goes through the center. This is the path along which the gears come into contact, and the actual torque transfer takes place. The slant of this slope can be selected to suit your needs, but it needs to be reflected in the geometry of the resulting teeth:



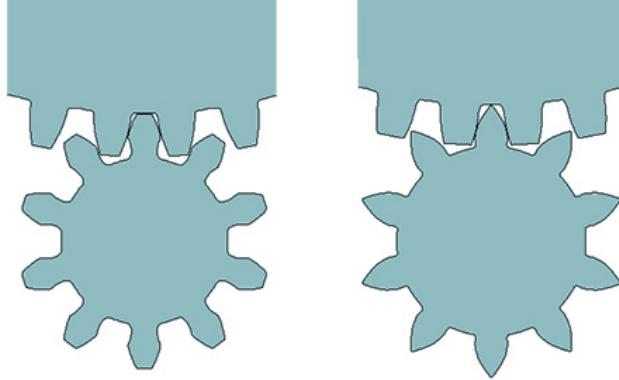
In general, low pressure angles result in quiet-running, precise gears that are easy to machine due to generous clearances - but will be weaker, and have more pronounced undercuts. High pressure angles result in thicker tooth profiles which need to be machined with smaller tools, and can be more noisy and prone to backlash and slippage - but also survive a lot more abuse.

Sensible values for this parameter range from 12° to 25° or so - and my personal favorite for miniature assemblies is 17.5°.

### 6.2.4. Addendum

The addendum is the distance by which every tooth sticks out above the pitch (contact) circle. The recommended value that ensures continuous and smooth torque transfer is  $\text{tooth\_width} * 2 / \pi$ , with some wiggle room if you are pressed for space or are running into other constraints. Going below 70% of the recommended distance should be done at your own risk.

To minimize undercuts when dealing with low pressure angles and low tooth counts, you can use a larger addendum on the smaller of the two meshing gears, and a smaller addendum on the other one. This technique, known as *profile shift*, is particularly useful for gears with fewer than 12 teeth, where it can significantly reduce undercuts without compromising machinability (right):



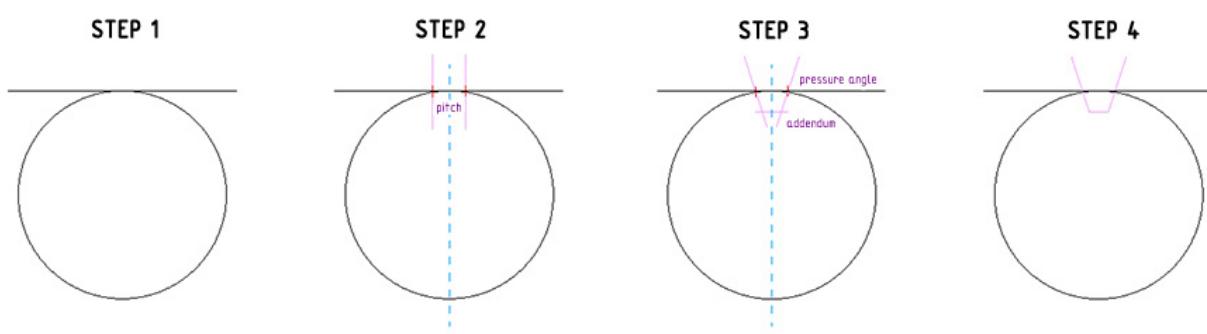
Of course, profile-shifted gears will not mesh correctly with regular ones - the mating gear always needs to have exactly the opposite shift. This is seldom a problem, but you need to keep this constraint in mind.

### 6.2.5. Putting it all together

All right! Equipped with all this knowledge, you can compute the profile for your first set of involute gears. There are many different approaches to this task, but my favorite method is unique in that it comes without the limitations that plague simpler techniques - and if you are so inclined, it can be easily generalized to deal with a variety of unusual gear shapes and tooth profiles later on.

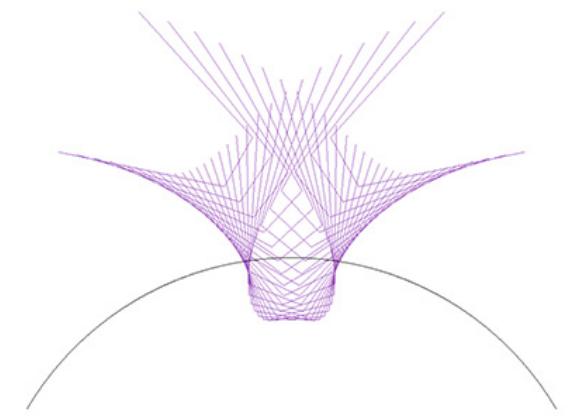
The process begins with calculating the circumference of the pitch circle for your gear:  $c_{pitch} = \text{tooth\_width} * 2 * \text{tooth\_count}$ . From this, you can trivially calculate its diameter or the circle you need to draw ( $c / \pi$ ). Once you have it on your screen, you need to add a straight line tangent to the pitch circle; this will be used to build a hypothetical mating gear with an infinite diameter. If we can mesh with this, we will be also able to mesh with any smaller gears encountered in real life.

On this "pitch line" of our infinite gear, we can draw two lines perpendicular to it, precisely `tooth_width` apart - and rotate them around the point where they intersect the infinite gear by an amount corresponding to the pressure angle we want to maintain ( $18^\circ$  in this example). To turn this into a proper tooth, we need to add a horizontal line that is normally placed  $\text{addendum} = \text{tooth\_width} * 2 / \pi$  below; in fact, draw the line about 5% farther down to create some clearance for any dirt or grease caught in between gears. When done, trim off the excess - and voila:

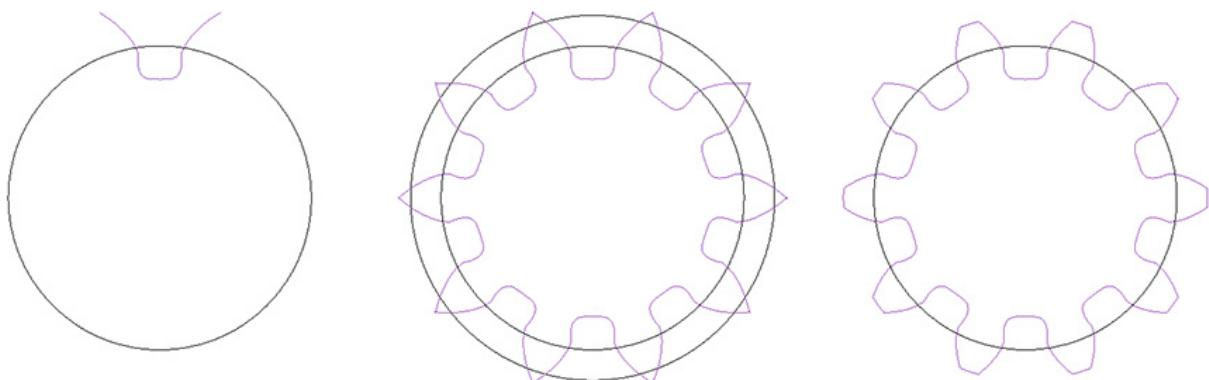


The next step is to simply perform a simulation of how this tooth would engage and then disengage with the actual gear we are trying to create. Recall the mention of rubber rollers: for any two meshing gears, the linear velocity at the pitch circle should be the same - so if the infinite gear (aka rack) moves by  $n$  millimeters, the other gear will rotate by  $360^\circ / c_{pitch} * n$ .

The rest should be fairly obvious; this is the pattern you will see if you keep creating copies of the tooth at successive offsets along the rack, and then rotating them back by a matching angle around the center of the pitch circle of our gear:



The process can be automated easily in almost every CAD tool, although for the initial try, it makes sense to do it by hand; in any case, once you have this pattern generated, you simply need to trace a curve around it, replicate this curve using the polar array tool to create the preselected number of teeth, and offset the pitch circle by the addendum distance to trim off the unnecessary bits:



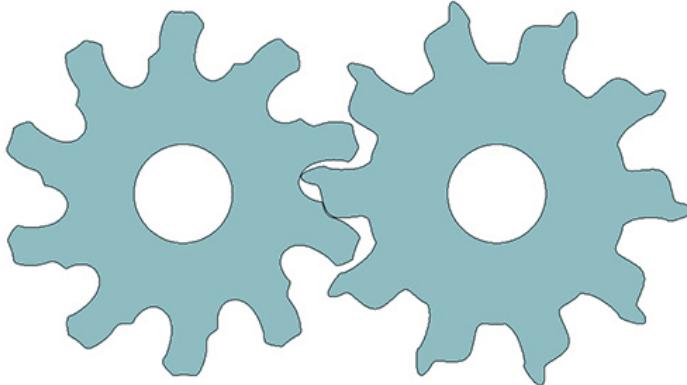
---

Try to do this exercise on your own. With rare exceptions where backlash is completely unacceptable, you should draw the outline about 0.05 mm apart from the trace left behind by the simulation; without this small amount of play, the teeth will engage on both sides simultaneously, producing unnecessary friction.

### 6.2.6. Bonus tricks

The beauty of the simulation-based method of constructing gears is that it's WYSIWYG, and that it takes most of the usual constraints out of the equation: for example, you can produce gears with any number of teeth, because the possibility of undercuts is automatically accounted for. On top of that, it can be modified in several fairly intuitive ways. You can:

- **Use custom tooth profiles:** instead of the trapezoidal tooth profile we created in our example, you can as well draw a puppy, and still get a pair of working, matching gears. It won't be particularly efficient, but it will work. I'm not good with puppies, but here's an example of a less cute freeform tooth and its generated counterpart:



More practically, you can also design gears that have an involute profile, but mesh only with a particular matching gear, by using that gear's tooth profile and its pitch circle, as opposed to a rack. This will help you reduce undercuts when other methods fail.

- **Create custom gear shapes:** there is nothing that forces you to perform the simulation against a pitch circle - the algorithm can be easily extended to use an arbitrary "pitch curve" to generate [oddball gears](#) that produce variable transmission ratios and do other funny things.
- **Draw internal gears:** "inverted" gears - essentially a ring with teeth pointing to the inside - are indispensable in certain types of mechanisms, including high-performance transmissions covered in section 6.4. These need to be generated to mesh with a specific gear running on the inside, using its tooth profile, and are tough to get right by other means.

Sounds more fun than it should be? You bet it does!

**Tip:** of course, there is no need to repeat all these steps by hand. The most flexible option is [this web-based generator](#) developed by Rainer Hessmer. The tool is very user-friendly: simply enter several parameters and get a DXF file in return. The tool properly handles undercuts, internal gears, profile shifts, and so on - so take it for a spin!

If you're using Rhino, you may also enjoy my experimental [Grasshopper module](#), which provides a comparable degree of flexibility, but is much faster, with instant feedback right within the CAD environment. It comes with fairly detailed instructions and features important for machineability, too. To run the add-on, you will need to install the [Grasshopper](#) plugin itself, open the plugin with the namesake command, and then load the [gear\\_calc.gh](#) file into the app. Note that the internal gear functionality is not fully operational yet: undercuts are not properly compensated for if the diameter of the meshing gear is close to that of the internal one.

Last but not least, some Rhino users may want to grab a copy of my earlier [semi-automatic script](#). Its main advantage is that it can handle arbitrary tooth profiles and very accurately handle internal gears. It is activated by entering [Hob4](#) in the command prompt and answering some simple questions about the gear you want to generate. It will then draw the appropriate circles and - if instructed to do - a standard trapezoidal tooth profile. You can modify the profile or simply select the one produced by the script, and run [Hob4](#) again to finish the job.

As noted, the script also allows you to generate internal gears; to do that, you must provide a mating pitch circle and tooth profile extracted from a previously generated external gear. This circle needs to inside, and must be touching the top of the pitch circle for the internal gear.

### 6.3. Other types of gears

Involute spur gears are remarkably versatile, easy to model, and easy to manufacture - but they are certainly not the only type of a gear you can use. Let's go over some of the more exotic but still useful options that may come handy in your work.

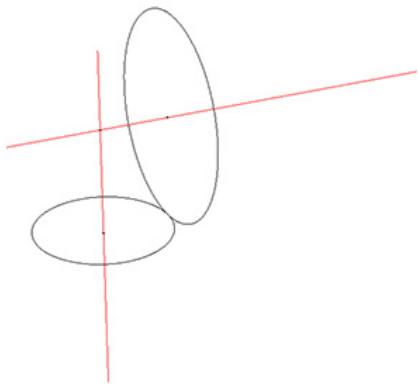
### 6.3.1. Bevel gears

In comparison to planar spur gears, bevel gears have a very important property: they mesh at an angle, and therefore, can be used to change the axis of rotation. Although this goal can be achieved by other means - for example, with simplified [crown gears](#) or with [worm drive](#), bevel gears do so with remarkable precision and efficiency.

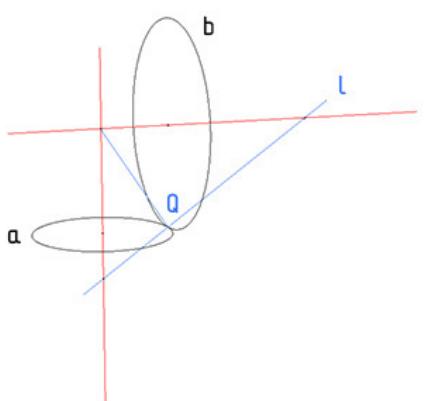
The process of constructing a set of meshing bevel gears is considerably more complicated than what we discussed before, and involves a brush with non-Euclidean spaces; nevertheless, if you follow these instructions closely, you should be able to pull it off.

As with spur gears, the first step is to choose tooth counts, width, height, and pressure angles; you should also think about the exact gear ratio you want to achieve, as one pair of gears will be not interoperable with any other differently configured set.

Next, draw the appropriate pitch circles for both of the gears you want to create, placing them on the X-Y plane; when done, add two lines perpendicular to this plane and going through the center of each circle. Finally, rotate one of the circles (and the associated perpendicular line) in the Y-Z plane, pivoting the operation around the point where touches the other one. In our example, we will use  $90^\circ$  between the mating gears, but any other angle is a possibility:



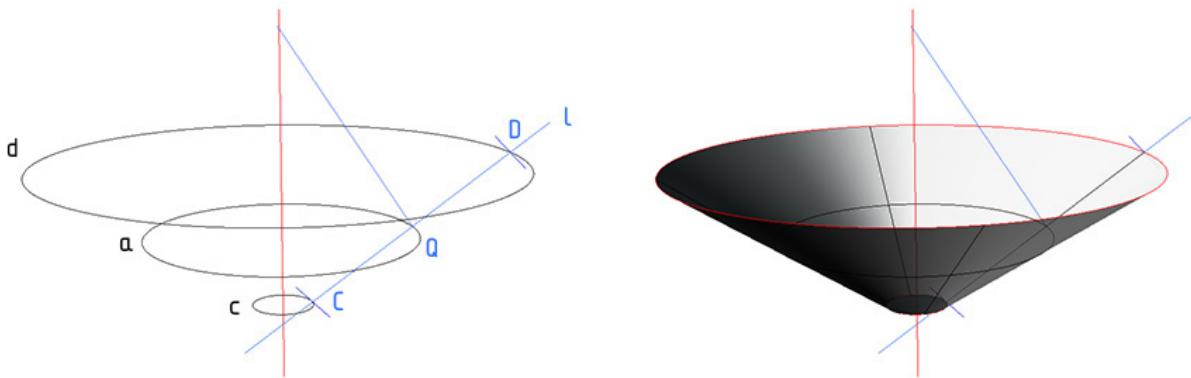
With this out of the way, draw a line that starts where the two perpendicular lines meet each other, and ends where the two circles touch each other (**Q**). Next, using the Y-Z drawing plane, add a line perpendicular to this newly created one. This final line is denoted as **l**:



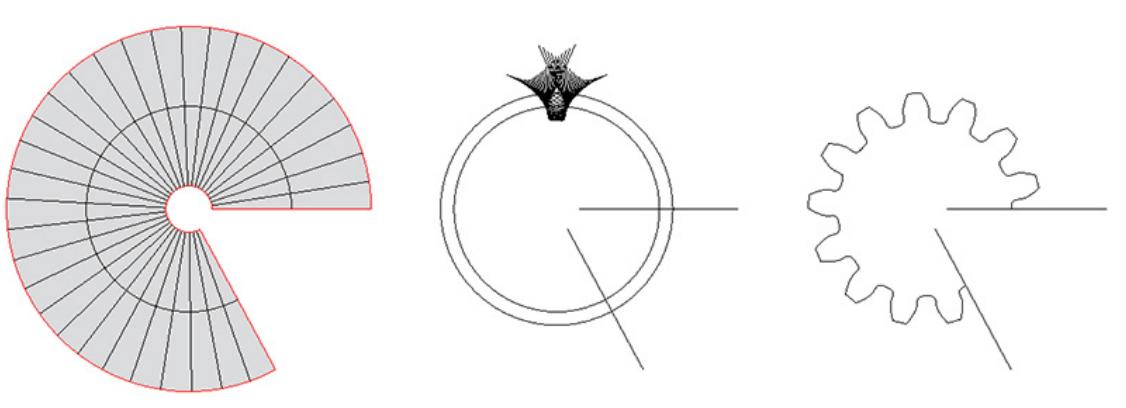
To minimize clutter, we'll focus on constructing the gear profile for circle **a**, and hide the other circle for time being; the steps needed to construct a suitable gear profile for the second gear will be analogous to what we are doing past this point, so you should have no difficulty figuring it out.

The next task is to mark two locations on the line **l** that are in equal distance from point **Q**; these newly created locations are denoted as **C** and **D**. The distance must be a bit greater than the addendum for your gears, but beyond that, the exact value doesn't matter. With these two points in place, draw a circle in the X-Y plane that has

the same center as circle **a**, and passes through point **C**; a similar circle should be drawn for **D**. Lastly, use the loft operator after selecting the newly created circles **c** and **d** to build a conical surface:



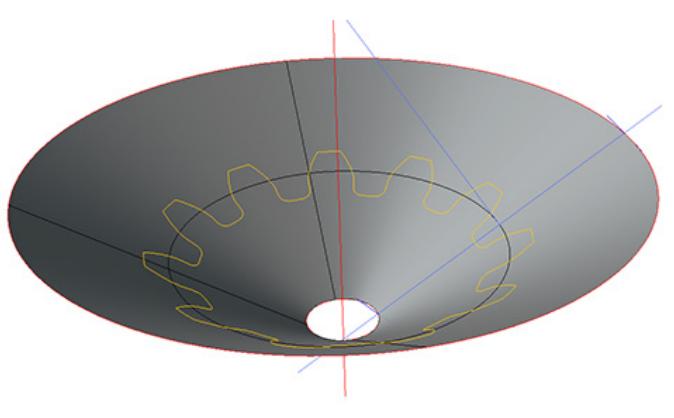
The next challenge is to check the manual for your CAD application, and identify an operator that allows you to "unroll" this surface to a construction plane; in Rhino, `UnrollSrf` is what you need. Select the surface and the original circle **a**, and apply this transformation; you should get an object resembling the one on the left:



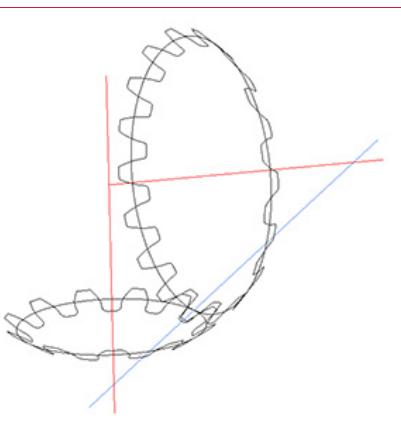
Extract the two "broken" edges of the surface as straight curves, and draw a circle that has the same center and diameter as the arc produced by unrolling the curve we selected alongside with the cone. This will be your *virtual pitch circle*, on which you will have to construct a regular tooth profile using the approach outlined earlier on.

You can start by simulating the appropriate tooth profile (center, above), and then by making a planar array to complete the gear (right). The only difference is that before making the array, you need to rotate the profile by 90° to touch one of the extracted edges; and then make a radial array consisting of `tooth_count` copies that fill an angle of  $360^\circ / \text{length\_of\_virtual\_pitch\_circle} * \text{length\_of\_circle\_a}$ , rather than the usual  $360^\circ$ . When done, truncate the parts that stick outside the edges, and you should be all set.

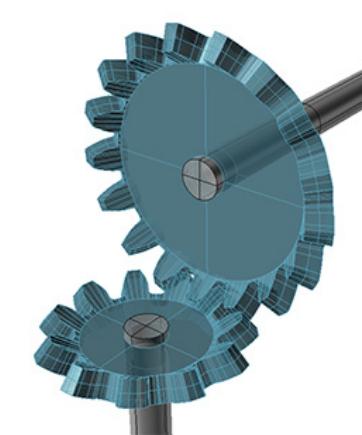
To wrap up this process, you need to position the created gear profile over the unrolled surface again, making sure that it is aligned correctly. Next, use an operator that will map the curve back onto the original conical surface, using the unrolled geometry as a reference. In Rhino, this is accomplished with `FlowAlongSrf`. Just make sure that you select matching locations on the reference and destination surface, and if you did it right, you should be able to get something of this sort:



You can delete the cone and all the helper surfaces at this point, and repeat the same procedure to generate the other gear. In the end, you should be seeing a result that resembles this:



Neat, eh? You can now create a scaled copy of each outline along the appropriate axis, using a 3D scaling operator with the origin set at the point where the axes intersect; be careful with this step, as using an incorrect origin will result in gears that would not mesh. With this step out of the way, the scaled copy and the original can be then lofted together, and each of them can be individually extruded toward any point behind the gear to create a cap surface. A few extra moves to trim and cap everything neatly, and you should have a final result in front of you:



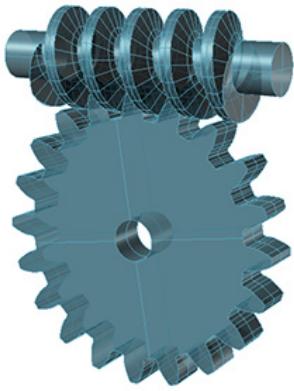
Sweet!

### 6.3.2. Worm gears

Worm gears are an interesting animal. On one hand, they offer extremely high reduction ratios, which makes them quite attractive for robotics: every turn of the input shaft advances the output gear just by a single tooth. On the other hand, they generally boast very poor efficiency due to significant friction - 30-40% being the norm.

From the design standpoint, these gears are fairly unremarkable; the output gear is generated the same way you

would make any other spur gear, while the input gear - or more accurately, the input screw - is essentially a rack tooth that is wrapped around a spiral path, with a `tooth_width * 2` pitch between the turns:

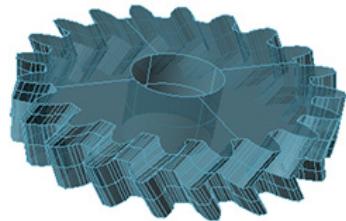
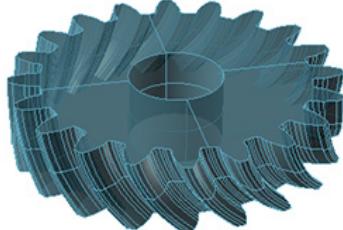
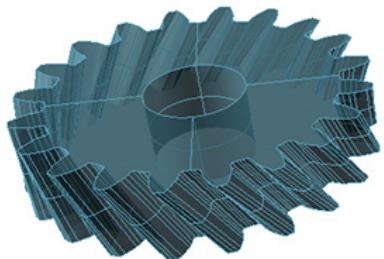


---

Depending on the thickness of the output gear and the diameter of the input screw, the profile used to generate the output gear may need to be corrected slightly, based on a 2D projection of the trimmed, meshing section of the screw. You should be able to create all this with no help by now; operators such as `Sweep1` and `Silhouette` in Rino are about everything you need. The only complication for CNC machining is that as opposed to spur and bevel gears, the input screw needs a two-part mold.

### 6.3.3. Helical gears

Helical gears are essentially spur gears - but (literally) with a twist. Instead of having their teeth extruded along the axis of rotation, the top and bottom profiles are rotated in relation to each other, and then lofted to form a twisted gear. The twisting may be done in a straight line (left) or in a curved fashion (center):



The point of this modification is that each gear will now engage more gradually, thus reducing the amount of noise and vibration produced under load, this matters particularly in automotive applications and other situations where steel gears transfer high torques at high speeds. On the flip side, this geometry means that the gears act like a wedge, and exert substantial axial forces on the assembly they are coupled to; a double helical layout, also known as *herringbone* (above, right) works around that to a large extent.

Helical gears are relatively difficult to make using simple, CNC-machined molds; for optimal results, the mold would need to be split into four to six parts. It is easier to machine them directly using a rotary axis - but in any case, the benefits of using this class of gears are negligible in small-scale robotics work.

### 6.3.4. Pulleys for timing belts

Gears that mesh with timing belts are constructed in a manner fairly similar to a normal gear meshing with a rack, but their shape is not entirely symmetrical. MXL timing belts use 40° teeth with a pitch of 2.03 mm. Addendum of the timing belt is approximately 0.5 mm; the dedendum is zero. The width of base of the tooth (at the pitch circle / line of the belt) is around 1.2 mm, leaving just around 0.8 mm in between the teeth. The solid section above the gears is usually around 0.7 mm thick.

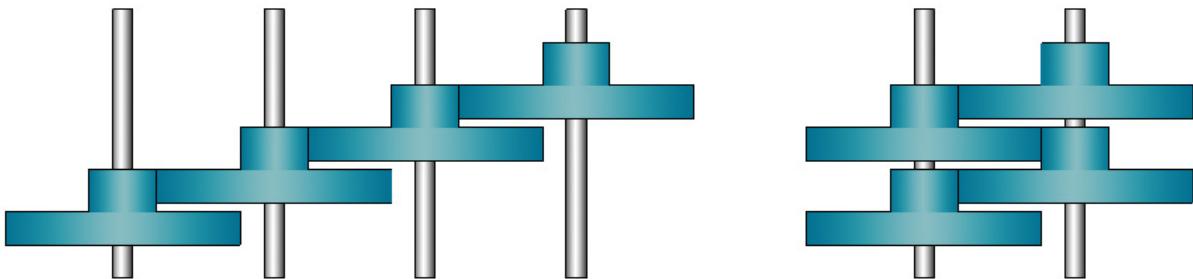
Of course, the belts are made of rubber and will easily conform to pulleys even if the dimensions are slightly off - so you don't need to obsess over it as much as you'd have to with rigid gears.

## 6.4. Getting started with gearboxes

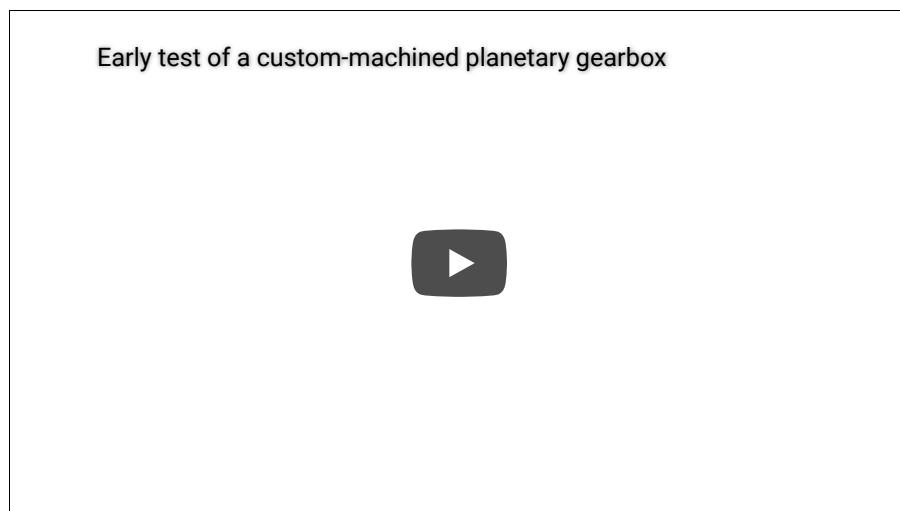
It is possible to adjust the speed or torque of a motor by employing just a single pair of gears, but doing so is often impractical: for example, if you have a motor that nominally runs at 10,000 RPM, and want to get down to 100 RPM - a reduction of 100:1 - you would need to mesh an input gear with about 10 teeth to an output gear with no fewer than 1,000. Assuming that the width of a single tooth is 0.6 mm, the diameter of the latter gear would be almost 50 cm - oops!

To solve this problem, it is common to build gearboxes that employ multiple stages; each stage reduces the speed by a more reasonable factor - usually between 2:1 and 5:1 - and then powers the next stage that performs the same trick again. The gains are exponential: if one stage gives us 4:1, then two will offer a 16:1 reduction; three will yield 64:1; and four would get us down to 256:1.

The most rudimentary type of a gearbox is the offset-gear arrangement shown on the left; you can check out a practical example of this design [here](#). This approach works well, but tends to take up space; a simple modification, shown on the right, is to stack the gears alternating between two shared axles, instead ([example](#)):



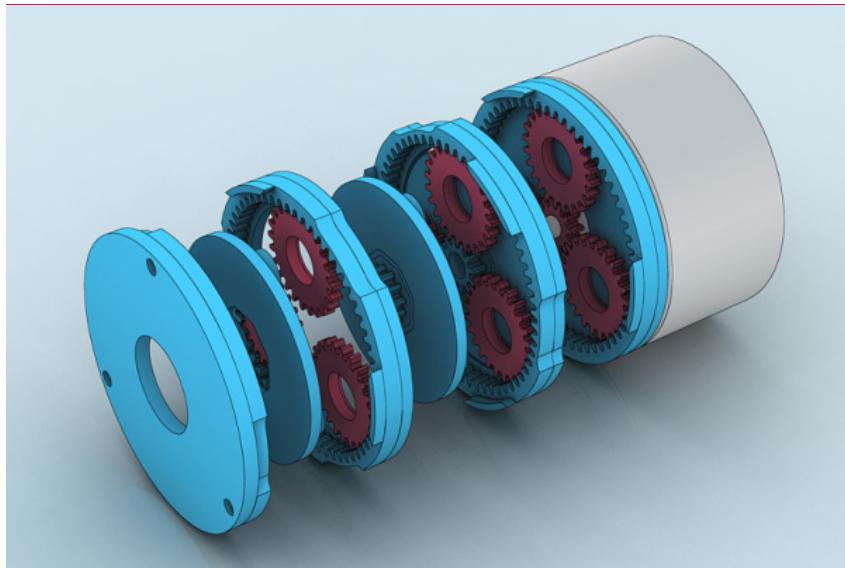
Offset gear transmissions of this sort are simple and adequate for most needs; for instance, [Tinybot mk III](#) uses them exclusively. Having said that, every now and then, you may wish for something more compact and self-contained. If you are pressed for space, it makes sense to consider a [planetary gearbox](#). There are several possible ways to design such a mechanism, but the most practical arrangement relies on a small, centrally placed *sun* gear coupled with the input shaft; that gear is surrounded by several *planets*, and these planet gears in turn roll inside a stationary internal gear known as *annulus* or *ring*:



Of course, this design makes little or no sense if the planets just run freely; the output stage is created by creating a carriage that slides onto the planet gears, often using a sleeve bearing. The transmission ratio of such a

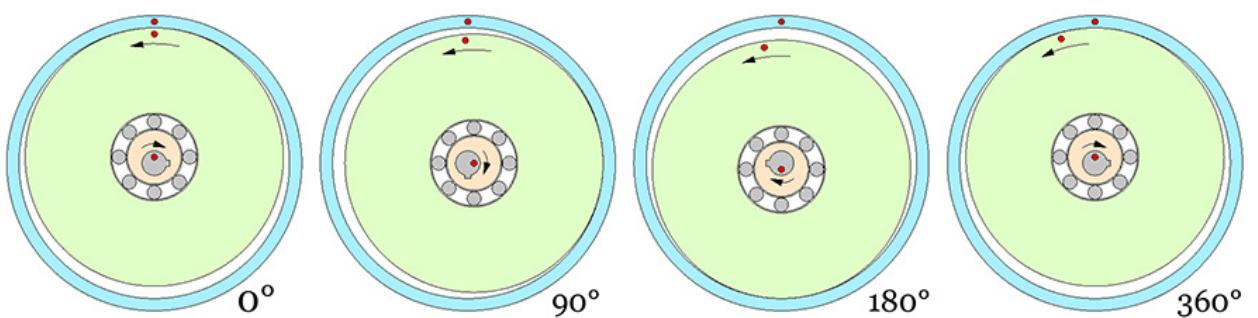
mechanism will be  $\text{tooth\_cnt}_{\text{sun}} / \text{tooth\_cnt}_{\text{annulus}}$  - and that's quite a good result in such a small envelope. Just as importantly, additional stages of a planetary gearbox can be packed tightly on top of each other, resulting in a form factor that is pretty hard to beat.

To better illustrate the practical aspects of making a planetary gearbox, you may want to check out this rendering of the design I have been using in my projects for [quite a little while](#):



Of course, planetary gearboxes are no silver bullet; they are about 2-3 times more compact than offset gears, and outperform them in some ways, but it's possible to run into situations where an even higher ratio would be desirable - but where the poor efficiency of worm gears and other types of screw-based actuators is hard to swallow.

Of all the more exotic transmission systems, the most promising one is probably the cycloidal drive. This device easily attains ratios of 100:1 or more in a single stage by brilliantly exploiting eccentric motion of a large gear in a comparably-sized "cage":



I have a [separate page](#) describing this particular design in more detail, so I won't repeat all the information here. Rest assured, cycloidal drive systems have some substantial downsides - for example, they tend to introduce vibration, and require subminiature ball bearings to isolate the input shaft from the eccentric gear. Nevertheless, when out of mainstream options, they are worth a try.

## 6.5. Overview of drive systems

By now, you should be well-equipped to get ahead with almost any electromechanical project you can think of; the only question is where to start. In the world of robotics, the single most important choice is probably the mode of locomotion; everything else - from the selection of sensors, to power sources, to data processing capabilities - follows from that. In the last section of this chapter, it makes sense to quickly run through the options you have.

In all likelihood, your initial projects should probably be wheeled. It is tempting to try something bolder, but until you are familiar with the dynamic properties of plastics, and have several precision robots under your belt, it's really the best choice. Some of the common approaches to wheeled robots include:

- **Differential steering:** in this design, the two front wheels are individually hooked up to two motors; the rear wheels are typically spinning freely. When the motors are rotating at the same speed, the robot drives in a straight line; speed differences result in the chassis banking to the left or to the right. The rear wheels typically need to be equipped with rotary encoders or optointerrupters to detect and compensate for unintended fluctuations in motor speed, and to sense loss of traction.

The main disadvantage of this design is that during sharp turns, the non-powered wheels will be dragged sideways, which may lead to loss of registration with the environment. The problem can be avoided by using swivel wheels - or even eliminating the rear wheels completely and lowering the center of mass so that the device is stable with just two; but all these solutions make it much harder to maintain a robust feedback loop. That last problem may be overcome by using [Mecanum wheels](#) - but the price tag may be hard to swallow.

- **Car-style steering:** in this arrangement, the front wheels are spinning freely; a rigid linkage operated by a motor allows them to turn left and right while a separate motor powers the rear wheels. Position sensors placed on the front wheels provide a feedback loop. This is the arrangement used in [Tinybot mk III](#).

The approach is simple and sweet, which is precisely why it's so popular in the real world. That said, not all is roses: for accurate, slip-free turns, you may still have to incorporate [Ackermann linkages](#) and a [differential](#) on the rear axle. This adds more complexity than you could be expecting.

- **Fully reorientable drivetrain:** this approach relies on a drivetrain with a system of linkages that allows every wheel to be repositioned to accommodate motion in a particular direction. A simple and economical design of this type, using only three motors and permitting on-the-spot 360° turns, is employed in my [Omnibot](#).

The only disadvantage of the specific approach used in Omnidroid is that the robot needs to come to a stop before changing direction. In situations where this is a significant problem, additional motors may be used to independently control the position of each wheel.

Wheeled robots aside, legged creatures have some undeniable appeal. These robot designs tend to pose a range of challenges related to space management, torque requirements, obstacle sensing, and the creation of practical movement algorithms. Sure - it is relatively easy to [bolt a bunch of expensive servos](#) on top of a simple frame - but going beyond that point is usually fraught with peril. Still, if you want to give it a try, your basic choices are:

- **Biped locomotion:** achieving a natural, stable, and useful walking gait probably requires at least 4-5 degrees of freedom per leg, packed in a very narrow envelope - as well as a bunch of gyroscopes, accelerometers, and contact or current sensors to detect problems while it's still possible to recover. Because of this, and because of the algorithmic complexity of dynamically stable biped motion, most of the designs you can see tend to cut a lot of corners; using oversized feet to achieve static stability, or executing turns in a large number of small and awkward movements, is a common pattern.
- **Quadruped and beyond:** with four or more legs, it is relatively easy to create a design capable of moving in any direction without the need for sophisticated stability control. At least three degrees of freedom per leg are probably required to achieve efficient walking gaits (making turns, climbing obstacles, etc), but some success on flat surfaces can be achieved with only two degrees. Hexapods seem to be particularly popular with hobbyists - perhaps because of their sinister look?

Of course, a number of designs for "scuttling" robots with one or less than one degree of freedom per leg can be found on the Internet - but they are usually not any more practical than a wind-up toy.

These movement modes aside, there is a long tail of exotic inventions that are worth at least reading about. We have hopping [pogo stick robots](#); several well-publicized [ball-balancing units](#); a bunch of levitating designs that leverage air cushions, magnetic fields, or vibration; insanely cool [crawling robots](#); and a large selection of flying and sailing ones. And hey - why not add something of your own to that list?

## 7. Workshop safety topics

---

We're done with the technical stuff, but stay with me for a bit longer: even for DIYers, workplace safety is no laughing matter. You will be dealing with power tools and reactive chemicals, and so if you want to pursue this hobby, there are inherent risks you simply have to understand and accept. It's entirely possible to lose an eye or set your house on fire; there are certain common-sense steps you can take to minimize the probability of mishaps, but the odds will never go down to zero.

It is also your sole responsibility to investigate and follow all the applicable HOA rules, city ordinances, and other regulations - especially as they relate to purchasing laboratory supplies and chemicals, the use of power tools, and the disposal of unreacted chemicals. In many places, the laws are sensible or non-existent; but [exceptions happen](#). If you accidentally ignore a piece of obscure, boneheaded legislation - well, nine out of ten times, nobody will know or care, but do you really want to be that one unlucky guy?

OK. With all that in mind, this chapter provides a quick but [non-authoritative](#) overview of some of the most obvious safety considerations and health risks for CNC machining and resin casting. There are no guarantees that this information is accurate, complete, or up-to-date, so do your own research. Do not blindly trust one random dude on the Internet!

## 7.1. Working around milling machines

Small CNC mills are fairly safe, compared to most other power tools - but you still need to read and follow the instruction manual, and in general, exercise common sense in everything you do. The primary risks are getting your hand pinched between the rotating tool and the workpiece, or between any other moving parts; having loose clothing or hair caught by the spindle; or being hit by any debris ejected during the cutting process.

Because of this, you should really stay clear of the machine while it's running a job - after all, you have no business fiddling with it. If you really need to make some sort of an adjustment, or perhaps simply vacuum off accumulated shavings, it's imperative to pause the cutting process first. It is also a very good idea to wear basic [eye protection](#) at all times; although the odds of a broken tool or other sharp material hitting your eye are very low, the damage can be significant.

All that aside, high-speed CNC lathes and high-power CNC mills (over 1 kW or so) have their own safety rules that I will not cover here. These devices are often powerful enough to break bones, sever limbs, or eject a loose workpiece with a speed sufficient to kill or maim. If you have one of these, the use of robust safety covers made out of sheet metal, polycarbonate, or wood, is definitely a must.

## 7.2. Dust considerations

Any type of dust can aggravate existing respiratory conditions; prolonged and heavy exposure may also lead to the development of brand new ones. That said, most of the noteworthy materials give off relatively little airborne particles when machined in a sensible way - and so, your overall exposure to nuisance dust will probably not change in an appreciable way.

When working with prototyping boards such as RenShape 460, airborne dust is produced in modest amounts, and has no known, specific health risks; the same goes for many other common plastics. Covering the mill with plastic sheeting for the duration of the job (well clear of any rotating parts and inlet vents) is a good practice, even if just to keep your workplace tidy; building or buying a proper enclosure is an option, too. In truly extreme cases, a simple N95 particulate respirator ([link](#)) is good to have at hand.

That said, there certainly are materials that pose an elevated risk; to avoid really unpleasant surprises, always investigate the substances you are milling, sanding, or otherwise dispersing in the air. You should be particularly wary of anything that contains crystalline silica (quartz), asbestos, elemental metals, toxic pigments, and any other harmful substances that may be absorbed through the lungs.

In essence, be sure to handle any powdered fillers and pigments with care, and do not mill, sand, or cut rocks, and mystery plastics of unknown composition - and you should be fine.

**Note:** while crystalline silica (quartz) is a significant danger when inhaled, there is no compelling evidence that amorphous glass products, such as milled glass fibers or 3M glass microspheres, share the same risk.

*You still don't want to breathe any of that stuff in, but these fillers should be safe to use in their intended way - and a tiny amount of dust should be OK.*

## 7.3. Vacuum and high pressure

Vacuum isn't particularly dangerous to work with, but you should be aware of the risk of implosion. Plastic vacuum chambers should be inspected for cracking or crazing, and should never be exposed to high temperatures, substantial quantities of solvents, strong acids and bases, and so on. Sketchy-looking or homebrew chambers can be secured with packing tape, shrink-wrap foil, fabric, and similar materials that will retain debris if things go wrong. Other than that, you have very little to worry about; oh, if you are working in a confined space, it may make sense to vent the vacuum pump to the outside, so that the relatively small quantities of gases liberated from the resin - as well as the fine mist of mineral oil from the pump itself - don't end up right under your nose.

Pressure casting is more dangerous; that's in part because the pressure differentials are much higher than for vacuum casting (especially if something goes wrong), and because all the energy is released outward if the container fails. Because of this, you should never try to rig your own pressure chambers or any other parts of the system, and not even think about removing any safety valves, regulators, and so on. Follow the instructions religiously, especially when it comes to draining the compressor; do not exceed manufacturer-provided pressure ratings; always double-check that the system is depressurized before trying to open the chamber; and disconnect everything completely when not in use. With common-sense precautions, catastrophic failure is unlikely - but rest assured, bad things have happened to quite a few people who should have known better.

Oh, one more thing: if you are inclined to get a nitrogen tank for blanketing resin containers in storage, go for a small one - capacity of 1 cubic meter (40 cubic ft) is more than enough. Store it horizontally or attach it to the wall to make sure it can't fall over, and always use a regulator.

## 7.4. Silicone resins

Platinum cure silicones are believed to be essentially harmless; both the siloxane resin and the catalyst are non-reactive and show virtually no toxicity in animal studies. You should always read material safety datasheets for the specific product you are using, but chances are, the only risk you have to worry about is that the material is sticky, and spills can be annoying to clean up (naphtha and other nonpolar solvents can help). Beyond basic workplace hygiene, no special precautions should be necessary.

Now, I don't recommend using condensation cure silicones, but if you decided to choose that option, you should be aware that they often use a small amount of dibutyltin dilaurate, dimethyltin dineodecanoate, or a similar tin(IV) compound, to catalyze the reaction. The substance is corrosive, and more troublingly, exhibits some chronic toxicity. It is present at around 5% in the catalyst component, and usually at less than 1% in the finished product - but since it will leach out in favorable conditions, you should probably not be using condensation cure silicones to make anything that is routinely worn or handled by humans, or that comes into contact with food. When mixing these resins, it's best to use latex or vinyl gloves; basic eye protection is not a bad idea, too. Storing the catalyst in a suitable cabinet and out of reach of children is important, too.

## 7.5. Polyurethanes

Polyurethane resins consist of several very distinctive components, and it's probably useful to discuss them separately. Here's what you can typically find in the MSDS:

- **Isocyanates:** polyurethane formulations used for prototyping normally rely on fairly non-volatile isocyanates, such as liquefied methylene diphenyl diisocyanate (MDI), hydrogenated MDI (HMDI / DMDI), or isophorone diisocyanate (IPDI); products where these isocyanates are already partly polymerized are common, and in this case, the prepolymer may be not listed in the MSDS at all.

In general, all these substances are fairly reactive, can be damaging to mucous membranes, and upon longer exposures, will irritate skin. Fortunately, they polymerize to form an inert substance soon after being exposed to water, alcohols, and many other common substances - so they pose little danger to the environment, and do not bioaccumulate.

Isocyanates tend to be pretty dangerous if inhaled - enough to earn some of the water-clear HMDI and IPDI formulations a nice "toxic" label in Canada and in the EU. Nevertheless, the extremely low vapor pressure of these chemicals - around one million times lower than water - means that if you are using them for casting reasonably sized parts at room temperature, you have very little to worry about. Within the scope of this guide, the main things to avoid is getting the substance into your eyes and spilling it in large quantities. Use eye protection, wear latex or vinyl gloves, and keep your workplace safe and clutter-free.

Now, as hinted earlier, several manufacturers resort to more reactive and more volatile isocyanates, such as toluene diisocyanate (TDI) or hexamethylene diisocyanate (HDI); I recommend avoiding these, but if you have no choice, really good ventilation, gloves, and safety glasses are absolutely a must.

**Note:** *a small proportion of industry workers frequently exposed to significant concentrations of isocyanates eventually develop hypersensitivity to the compound. Although this is very unlikely to happen when working with room temperature casting formulations, you should obviously seek medical attention if you have difficulty breathing within several hours of working with this or any other exotic chemical.*

- **Polyols / polyetheramines:** most polyester or polyether polyol compositions are viscous, non-reactive, non-volatile fluids with relatively high molecular weight, chemically related to substances such as polypropylene glycol, glycerine, xylitol, and so on. They tend to have rather low toxicity in their intended use, and aren't particularly reactive. Not all products are the same - but unless something stands out in the MSDS, you probably don't have to worry (of course, common sense still applies).

Some systems may also use lower molecular weight amines as crosslinkers, or higher molecular weight polyetheramines to replace traditional polyols. Low molecular weight amines tend to be a bit more biologically active and irritating; a chlorine-containing amine originally used in many polyurethane formulations - 4,4'-methylenebis(2-chloroaniline) - was found to be a probable human carcinogen. A number of less reactive replacements emerged since then, and are expected to be much safer based on their chemical structure and animal data. In other words, there is no reason to panic, but it definitely makes sense to handle the formulations with some respect. A good sign of high amine content is a yellow-to-caramel color of the polyol component, alongside with a slight but disagreeable smell.

- **Catalysts:** these substances are of special interest, because as opposed to the core reagents outlined above, they usually remain as-is in the cured product - and if the circumstances are right, may leach out of it in small quantities, particularly out of Shore A elastomers.

Modern polyurethane chemistry is usually catalyzed with much less than 1% of a carboxylate of bismuth, zinc, tin(II), or a related metal; or with a handful of reasonably selective amines, imidazoles, and so on. Most of the contemporary metal-based catalysts are extremely unlikely to be of any concern - especially at the concentrations used in the resin. The non-metal catalysts have also been studied extensively and carry no known risks at these levels - but as noted earlier, they belong to large and interesting families of bioactive compounds.

As a relic of bygone days, several manufacturers still resort to toxic or bioaccumulative organometal catalysts, such as mercury neodecanoate or dibutyltin dilaurate; for example, that's the case for many products from Smooth-On and Freeman. In theory, the cured plastic should be safe to handle under normal circumstances. In practice, the mobility of the organomercury compound may be increased by factors such as incorrect mixing proportions, UV degradation, exposure to elevated temperatures, or the action of solvents and household chemicals.

In the end, I certainly wouldn't use the latter class of resins for jewelry, toys, or food applications - and I wouldn't want to worry about the exposure due to spills and other workshop accidents. I generally recommend staying away from such products altogether, especially since the alternatives have gotten much better over the years. It's not that mercury-catalyzed products are likely to maim

you, but they introduce a small, preventable, incremental health risk. When in doubt about the catalyst used, simply check the MSDS to confirm.

**Note:** if you are purchasing any concentrated catalysts or inhibitors to modify the performance of your resins (as discussed in [section 4.4.4](#)), be sure to do your homework. For example, common bismuth, zinc, and tin(II) carboxylates are believed to have very modest acute toxicity and no major chronic risks. Many other organometal catalysts - for example, based on tin(IV), cobalt, or mercury - are something you really don't want to keep around in concentrated form. Triethylenediamine and its friends sit somewhere in between: they are flammable, irritating, and fairly reactive, but probably not particularly toxic per se.

- **Plasticizers, fillers, surfactants, etc:** while most resins don't contain many other questionable additives, it's useful to briefly scan the MSDS for anything else out of the ordinary. For example, several manufacturers add phthalate plasticizers - such as butyl benzyl phthalate (BBP) or dibutyl phthalate (DBP) - to some of the more flexible polyurethane rubber formulations. While these substances have very low acute toxicity and were not shown to be conclusively harmful to humans, animal studies raise some worries about long-term subchronic exposure through children toys, food-contact items, and so on - a situation vaguely reminiscent of the controversy around bisphenol A (BPA). No need to panic, but educate yourself and decide what's your tolerance for such peripheral risks.
- **Residual solvents:** some resins may contain small quantities of solvents that are left over from the manufacturing process or have been intentionally added to help with mixing or degassing. This may include xylene, toluene, ethylbenzene, etc. Although the solvents are harmful upon prolonged, heavy exposure, the amounts present in the material are usually tiny (under 0.5%) and probably do not pose any serious concern.

As with any chemicals, you should avoid mixing polyurethanes - and isocyanates in particular - with potentially incompatible substances; as hinted earlier, this includes alcohols, water, but also strong bases, oxidizers, amines, and metal salts. Rapid polymerization, decomposition, or other funny developments may ensue. Isocyanates also decompose in a matter of minutes in dimethyl sulfoxide (DMSO), so do not use this solvent as a carrier for dyes, catalysts, and so forth.

In the same vein, don't mix unreasonable quantities of polyurethane resins in a single batch. Runaway exothermic reactions can become dangerous if the resin gets so hot that the mixing container melts away, or something catches fire. If you need to dispose of a large quantity of unwanted resin, polymerize it in batches; or at the very least, use a large, shallow pan and work outdoors, away from other humans and flammable stuff.

Last but not least, because of their irritant properties, uncured resins should be stored safely, and kept out of reach of children and other creatures other than the intended user.

These considerations aside, cured polyurethanes are one of the safer, more stable, and least controversial plastics out there. Even if you mess up the mixing ratio, the unreacted isocyanates will simply eventually react with moisture, unreacted polyols will not pose a threat, and nothing horribly bad should be leaching out of the material otherwise. Like most plastics, and organic materials in general, polyurethanes release a fair amount of toxic substances during thermal decomposition - carbon monoxide being by far the most significant problem - so try to resist the urge to burn them if at all possible.

## 7.6. Other chemicals

Throughout this guide, I have mentioned quite a few chemicals that won't be covered in this chapter in special detail. If you plan on getting any of them, you should take care to obtain and read the appropriate material safety datasheets, and understand all the associated risks. In no particular order, here are some of the things you should know:

- **Epoxy resins:** many of these products use corrosive and somewhat volatile compounds. Chronic toxicity may be a concern with daily exposure. When mixing larger quantities, beware of significant

exotherm - it's usually more pronounced than in polyurethanes.

- **Polyester resins:** styrene is highly volatile and flammable, and its vapors are harmful. On top of that, the standard catalyst is a strong oxidizer, and the resin typically cures with an extremely pronounced exotherm.
- **Pigments and dyes:** toxic pigments based on lead, mercury, arsenic, cadmium, chromium, and nickel are available on the market, but should be avoided. Shop specifically for modern, less harmful alternatives. Avoid pigments that contain crystalline silica of respirable size - and in any case, handle the material with care.
- **Glass fibers:** thin strands of glass are easily embedded in the skin, and can give you an annoying but generally harmless rash that takes 2-3 days to go away; getting the material into eyes can be a more serious issue, so use gloves, clean up promptly, and don't touch your face until then. If you get a rash, scratching it will usually only push the fibers deeper; soaking the affected area in water and applying an over-the-counter anti-itch cream (antihistamine) will offer relief.
- **Common solvents, glues, demolding agents:** unless indicated otherwise, you should assume that all these products are extremely flammable and that their vapors are harmful. Prolonged exposure to very high levels of certain solvents may lead to CNS effects, reproductive toxicity, and so on; over time, lasting damage is a possibility, too. Naphtha, xylene, toluene, benzene, methylene chloride, and NMP are of special concern.
- **All the other stuff:** several substances mentioned in this guide may be described as "reasonably safe", but that doesn't necessarily mean they are non-toxic if aerosolized or ingested in significant quantities; that they don't react in funny ways with incompatible materials; or that they won't cause damage if rubbed into your eyes or skin, heated up, or mishandled in some other way. Before buying any chemicals on eBay, Amazon, or anywhere else, look up their safety datasheets first.

**Tip:** MSDSes are targeted at industrial uses, and err heavily on the side of caution; the datasheet for table salt says that the substance is "harmful in case of ingestion"; that skin contaminated with it must be flushed with water for "at least 15 minutes"; that protective clothing and gloves should be worn at all times; and that proper ventilation is necessary. They also warn that you shouldn't let this chemical enter the environment.

And that's not even the worst example: Sigma-Aldrich has a serious datasheet for pure water that says you should use proper glove removal techniques and... wash hands after handling this dastardly chemical. To fend off lawsuits, they also make sure to mention that the chemical, physical, and toxicological properties of this material have not been thoroughly investigated.

Do not ignore the advice provided in these documents, but use common sense when evaluating your personal risk, and the safety measures that are appropriate given the amounts you are working with, the way you are handling the substance, the frequency of exposure, and so on. It's always good to pay close attention to LD50 values; chronic health risks; and any mentions of carcinogenicity, developmental toxicity, etc.

It hopefully goes without saying that you should store all the materials discussed in this guide away from sources of ignition, excessive heat, and so on; that you need to keep them away from children; and that if you are pregnant or breastfeeding, you should avoid any non-essential exposure to exotic, industrial chemicals - no matter what the MSDS says.

## 8. Closing notes

Well, that's it! I'm hoping that the content included here inspired you to some extent, or helped you solve a specific problem or two. Since this guide is incredibly niche, if you made it this far, your feedback would be greatly appreciated. If you are so inclined, drop me a mail at [lcamtuf@coredump.cx](mailto:lcamtuf@coredump.cx).

In any case, good luck in your future work!

*Your lucky number: 20607198*