Using a Self-Reversing Tapping Head on a CNC Mill

by Tim Paterson

Photos and Figures by Author

1

The Accusize self-reversing tapping head.

When I bought my manual mill/drill just a few years ago, all I wanted was a drill press with digital read-out (DRO). I was looking for the ability to both drill precisely positioned holes and also be able to return to the exact spot of a previous hole. The idea was to drill a set of holes the same size, then go back and tap them. Without the DRO, it would be drill-tap-move, drill-tap-move, constantly changing tools to ensure I didn't lose the hole position.

I started out just manually tapping holes using a spring-loaded tap guide. But to speed things up, I eventually shelled out for a self-reversing tapping head (Accusize 2600-4002, almost $300 on Amazon). Man, that baby cooks! In and out in just a few seconds instead of taking a minute by hand.

How It Works

A self-reversing tapping head is shown in ***Photo 1*** with a tap installed. It has three distinct mechanisms:

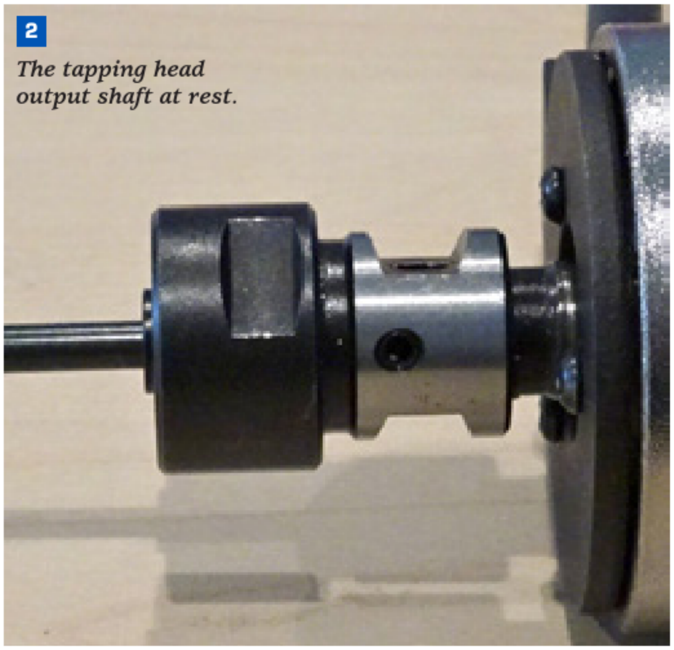
- Spring-loaded expansion/compression
- Clutch to limit torque
- Reversible planetary gear

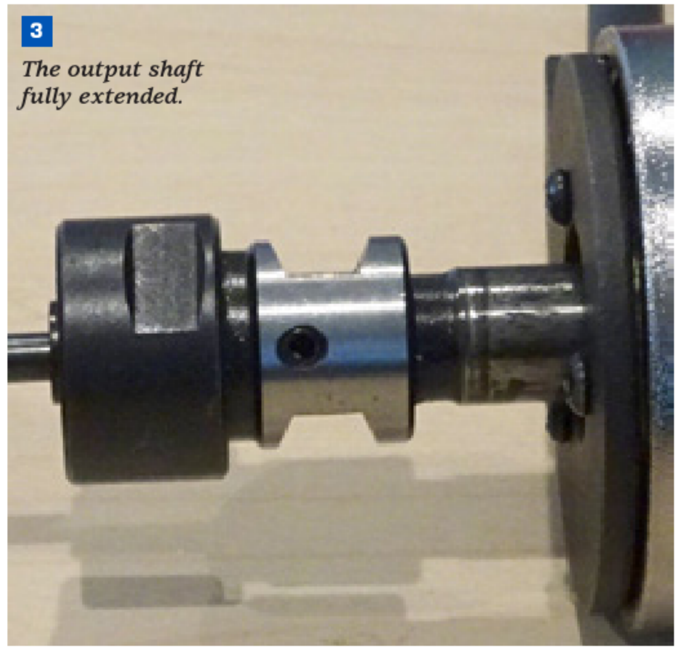
Photos 2 and ***3*** show the output shaft at rest and at full extension. Mine can move about 1/2" end-to-end.

The large, knurled black ring around the upper body adjusts clutch friction. I have mine set to minimum torque, yet it still managed to snap a 4-40 tap.

In order for the planetary gear drive to work, it must have a stationary reference – thus the 5" long arm sticking out sideways. You have to provide something to keep that arm from spinning around with the body. I use an indicator holder with a magnetic base.

Planetary gear drives are magical things. One of its simplest operations is pass-through – output RPM equal to input RPM. This is how the tapping operation starts. Then, as the tap cuts into the workpiece, the output shaft


2 *The tapping head output shaft at rest.*


3 *The output shaft fully extended.*

extends out. Once this self-feed has extended a specified amount (.20" on mine), the gear changes to neutral: the output shaft comes to a stop while the input keeps spinning, driven by the mill/drill.

Then an upward tug causes another gear change – into reverse. Due to the nature of the planetary drive this can't be at the same RPM as the input shaft. On mine, it's 1.6 times faster. (On a Tapmatic Rx, it's 1.75 times faster.) This reversal causes the tap to quickly unscrew itself from the workpiece. Once it's fully extracted, the head shifts gears back to the starting point, output equal to input, and is ready for the next hole.

Tapping a through-hole could hardly be easier. Lower the quill until the tap bites, then keep light feed pressure and follow the tap down. Stop the quill feed and allow the tap to continue feeding down until the gear shifts to neutral – at that point the tap won't be turning, so the feed will stop. Then, raise the quill and the tap will quickly extract. The amount of extension determines what gear the planetary drive is in. When you pull up, the increased extension is what shifts the gear into reverse. Since it goes out faster than it goes in, you might not keep up with the extraction rate. In that case, you'll hear the drive chatter a bit as it shifts back and forth between neutral and reverse. I can just let go of the quill feed lever and its gentle upward force keeps the quill tracking the tap.

The collet holding the tap is rubber and allows a little bit of play in aligning the holes (like .005"). This means that you should be able to relocate the hole accurately enough to go back and tap it even after removing the workpiece. And you can re-tap the same hole without hurting anything, including the workpiece. The tap will find the existing threads and follow them down. This allows you to extend the tap depth if it was done too short, or just let you practice on the same hole.

It's important to use the right kind of tap. The classic hand tap you find at the hardware store is not designed for continuous feed like this. You need a spiral point tap, also called a gun tap, which pushes the chips forward to fall out the bottom of the through hole. If you're doing a blind hole, then you need a spiral flute tap (sounds similar, looks way different), which pulls the chips up out of the hole. Tapping a blind hole with the tapping head on a manual mill/drill requires care in setting the depth stop and I never tried it.

On to CNC

When I bought my machine, I was just thinking about drilling. But since I had a mill/drill, I came up with some milling projects. And then some more. These were mostly just ideas and if I was going to get them done, I needed to move up to CNC. It only took about a year for me to jump in all the way and get a Tormach 770M.

There are tapping attachments specifically for CNC mills. An example is a tension/compression tapping head, which basically provides only one of the three mechanisms I listed for the self-reversing tapping head. It requires the mill to reverse the spindle rotation to stop further tapping and retract the tap, which some small CNC mills can't do (like the Tormach PCNC 440). This kind of tapping head can cost less than a self-reversing one.

But, I already had a tapping head, so I had to figure out how to use it on the CNC mill. There are no G-codes for "light pressure" and "let go of the quill feed lever." I needed to figure out what code to use for each of the three phases: feed, hold, retract. As an example, I'll use a 1/4-20 tap at 600 RPM, threading 1/2" deep; the tapping head has a self-feed of .20" and a reversing factor of 1.6.

Phase 1, the feed, is just a G01 move, of course, and the tapping feed rate is really pretty simple to figure. For each revolution of the spindle, the tap goes down by the thread pitch. For example, when tapping a 1/4-20 hole, the tap will go down 1/20" (.05") for each revolution. So, at 600 RPM, the feed rate for 1/4-20 is 600/20 inches per minute (IPM), or 30 IPM. And the feed must stop short by the amount of the tapping head extension.

Assuming Z0 is the top of the hole and we're already in position to start feeding:

```
G01 Z-0.3 F30.0
```

The hold phase would be G04, the dwell command. The dwell time should be enough for the tapping head to extend to neutral: dwell = extension/feed rate. But, with feed in IPM, that result would be in minutes, so multiply by 60 to get seconds. The result for this example with a 1/4-20 tap and .20" of self-feed: dwell = .2 / 30 * 60, or .4 seconds.

```
G04 P0.4
```

(Note: I have a generic G-code reference manual that says the dwell time is in milliseconds. That does not seem to be true for most machines, especially if a decimal point is present. Make sure you know your machine's behavior before programming.)

The retract has to go faster than the feed to account for the geared-up reverse. Just multiply the reversing factor by the original feed. In this example, 30 IPM * 1.6 = 48 IPM.

```
G01 Z0.6 F48.0
```

That line of code has a higher retract height than normal. This is to ensure the tap is always all the way out of the hole before moving on, even if the tapping head is at its full extension. I did learn this the hard way!

Let's put this all together, throw in an X and Y position, and set the spindle speed and coolant:

```
S600 M3 M8
G00 X-0.2 Y-0.1875
G00 Z0.6
G01 Z-0.3 F30.0
G04 P0.4
G01 Z0.6 F48.0
```

There is one more consideration I saw on a Tapmatic data sheet. If the mill has manual override adjustments for feeds and speeds, they need to be disabled for the duration of the tapping cycle to ensure the speed and feed stay in proper proportion for the thread pitch. This is done with M49 and M48, resulting in:

```
S600 M3 M8
G00 X-0.2 Y-0.1875
G00 Z0.6
M49
G01 Z-0.3 F30.0
G04 P0.4
G01 Z0.6 F48.0
M48
```

Picking Some Numbers

My first CNC tapping operation was using a 4-40 tap in 6061 aluminum, shown in ***Photo 4***. The recommended surface speed (again from a Tapmatic data sheet) was 50 SFM. According to my hole chart, a 4-40 tapped hole has a major diameter of .112". Since RPM = SFM * 12 / (D * pi), that comes to about 1,700 RPM.

Note that the recommended surface speed is independent of hole size. I actually got away with a few holes at this RPM before the tap snapped off in the workpiece. Not only did I lose the tap, but I also lost the work. There was just too much torque for that skinny little tap.

My conclusion is to always go slower. I reprogrammed the speed down to 550 RPM, nearly as low as the Tormach will go (on high range – I'm not changing belt position just to tap). That works fine for the 4-40, and the tapping speed is still incredible – about 1.5 seconds round trip (for .5" depth), including the extra feed height. And it just gets faster as the pitch gets coarser. Now I always use 600 RPM or less regardless of tap size. Lately, I've been picking the RPM to be a multiple of the threads per inch because I noticed the CAM was rounding the feed rate to one decimal place (more on that coming up). Using 560 RPM for 4-40 makes the feed rate a nice even 14 IPM.

Earlier I mentioned retract height; now let's talk about feed depth. Taps have a chamfer, so the cutting thread at the tip only cuts a little, while succeeding threads each take a bigger bite. On a through hole you typically use a spiral point tap with a plug chamfer, which has up to five incomplete cutting threads. You need to add this to your cutting depth to make sure you have fully cut threads all the way through. For extra margin, I add a depth equal to six threads. On a 4-40 this would be 6/40 = .15", in the 1/4-20 example it would be double that, 6/20 = .30".

The tapping head all set up in my Tormach mill. 4

When doing blind holes you use a spiral flute tap, and they come with a shorter chamfer to get full threads deeper into the hole. Mine are a modified bottoming chamfer, with 2 to 2.5 incomplete cutting threads. I add a depth of three threads to the depth my screw will need, which is exactly half the amount for a through hole. You must also make sure you drilled deep enough to account for this, including adding depth for the drill tip. You don't want the tap to bottom out in the hole.

I also mentioned that my Accusize tapping head has .20" of extension before shifting into neutral. The data sheet for this product claims this self-feed is .140" ± .004". I took a video of the tapping head in action to verify my understanding of how it worked. By knowing the RPM and counting video frames, I can say for certain that my unit extends .20".

Integration with Fusion 360

I use Fusion 360 for CAD and CAM and wondered, was there a way to get Fusion 360 CAM to generate this tapping cycle?

Fusion 360 comes with over 100 post processors ("post" for short) to handle output for different machines and combinations of features. I use the Tormach post, tormach.cps, which happens to have support for reversing tapping heads designed for CNC machines. The difference between that kind of tapping head and the one I have is they are not intended to dwell and extend – the CNC program

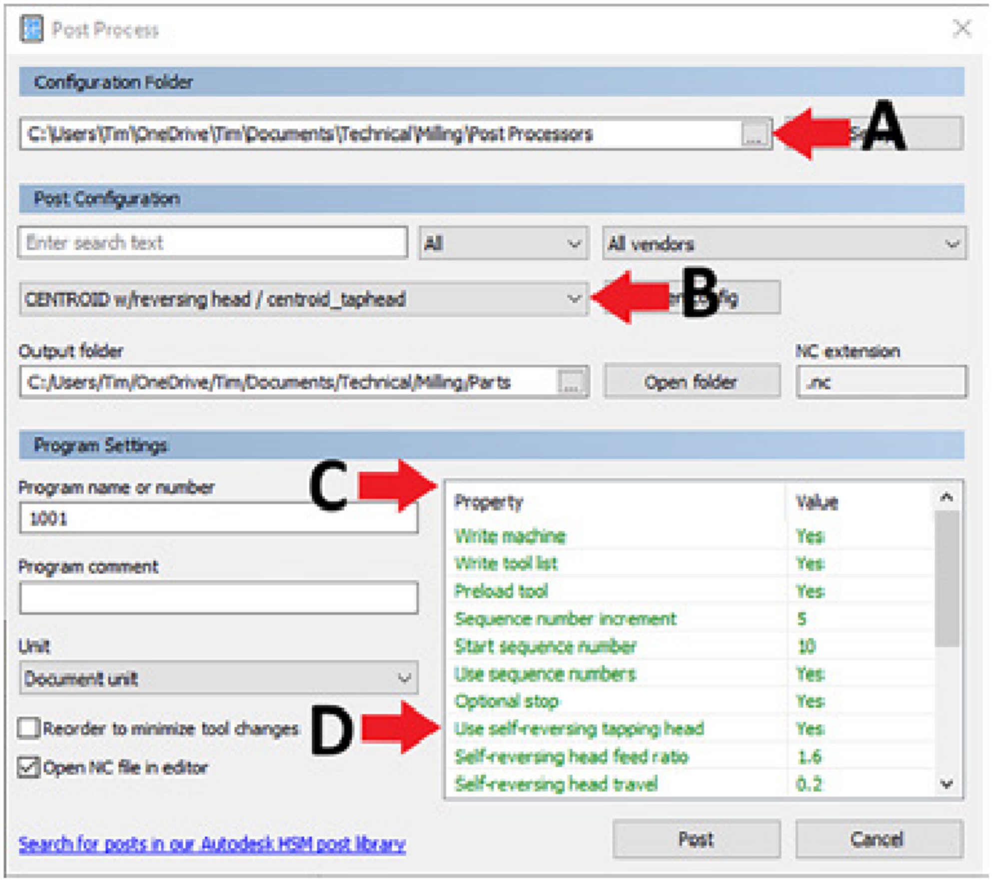


Figure 1: Fusion 360 post processor dialogue. Called out are: A – folder of post processors; B – selected post processor in drop-down list; C – property table; and D – properties added for the self-reversing tapping head.

immediately retracts after feeding to the programmed depth. However, starting with this level of support it was fairly easy for me to figure out how to make the necessary adjustments.

Even though these CNC tapping heads are available for any machine, it seems the Tormach post is the only one with built-in support. But now that I know how it's done, I can walk you through what changes to make to the post for your machine to add tapping head support. I took a look at posts for some of the machine controllers advertised in *Digital Machinist* (including Centroid, Masso, and Eding) and these changes should work with any of them.

```
description = "Tormach PathPilot w/reversing head";
```

Listing 1: Change the description. This is the first line of code below the header comment block.

```
// user-defined properties
properties = {
  reversingHead: true, // uses self-reversing tapping head
  reversingHeadFeed: 1.6, // percentage of tapping feed to retract the tool with reversing tapping head
  reversingHeadTravel: 0.20, // head self-feed
  …
  <Tormach: delete two duplicate lines>
};
```

Listing 2: Adding three new properties at the start of the list, near the top of the file. Set these default values to match your tapping head. Tormach already has the first two lines at the bottom of the list, so just delete them from there.

```
// user-defined property definitions
propertyDefinitions = {
  reversingHead: {title:"Use self-reversing tapping head", description:"Expanded cycles are output with a self-reversing tapping head.", type:"boolean"},
  reversingHeadFeed: {title:"Self-reversing head feed ratio", description:"The reversing gear ratio for retracting the tool.", type:"number"},
  reversingHeadTravel: {title:"Self-reversing head travel", description:"The tapping head self-feed after machine feed stops.", type:"number"},
  …
  <Tormach: delete two duplicate lines>
};
```

Listing 3: Setting the title and description (tooltip) for the new properties, just below the Listing 2 block. Again, Tormach already has the first two lines at the bottom of the list, so just delete them from there.

The first step is to make a copy of the post you use, maybe to a location easier to find than the folder where Fusion 360 keeps them. To find the folder, just run the Fusion 360 post command and you'll see the name at the top of the resulting dialog box labeled "Configuration Folder" – see ***Figure 1***, callout A. I used copy and paste to put the folder name into Windows Explorer, which I then used to find my specific post (tormach.cps) and copy it into a dedicated subfolder under my documents. I actually made two copies, just so I'd have the original handy. I named the file to be modified, tormach_taphead.cps.

The post is in JavaScript. Any text editor can be used to make changes. ***Listings 1-6*** show the changes that need to be made using highlighting. The text that is not highlighted is already present and shows you how to locate the spot in the file. You see an ellipsis (...) in a few places where there's a chunk of code that's not shown (generally because it's different for each post). There are special changes needed for tormach.cps, as noted in the captions for each listing.

Modifying the JavaScript code can be tricky only because the slightest mistake will cause it to fail and you won't have any idea why. I suggest running the post after every change or two to see if it's still working. To use the modified post, first you must point Fusion 360 at it. In the post dialog box, change the configuration folder (callout A in ***Figure 1***) to the folder where you put the new post.

Then, under Post Configuration, look at the post drop-down, ***Figure 1*** callout B. The drop-down will automatically be filled with a list of all posts in that folder, showing the description field. Select your modified post from the drop-down.

If the property table – callout C – is empty, that indicates a syntax error that's keeping the code from running at all. That could be something like a missing comma or unmatched brace. If that's working, go ahead and try the post on something simple. Check the G-code and see if everything looks right.

Let me explain a little about what the code changes do: First, they add three new properties, which are values you can edit in the Fusion 360 post dialog box. You can see them in ***Figure 1***, callout D. For example, you could change "Use self-reversing tapping head" to "No," and the post would generate the same G-code it did before making the changes.

The meat is in the new function **`expandTappingPoint()`**, which

```
function expandTappingPoint(x, y, z) {
  onRapid(x, y, cycle.clearance);
  writeBlock(mFormat.format(49)); // disable feed override
  onLinear(x, y, z + properties.reversingHeadTravel, cycle.feedrate);
  // dwell while head extends
  var dwell = 60 * properties.reversingHeadTravel / cycle.feedrate;
  if (dwell != 0) {
       writeBlock(gFormat.format(4), "P" + secFormat.format(dwell));
  }
  onLinear(x, y, cycle.clearance, cycle.feedrate * properties.reversing
HeadFeed);
  writeBlock(mFormat.format(48)); // enable feed override
}
function onCyclePoint(x, y, z) {
```

Listing 4: Find the function `onCyclePoint()` and add this new function above it. The function already exists on Tormach, but it needs to be replaced with this code.

does exactly the same calculations I did manually when creating the G-code example. The dwell time is in seconds, so if you have a machine that uses milliseconds you'd need to add code to multiply it by 1000. Note that one of the lines of the function adds "reversingHeadTravel" to the final Z position to account for the tapping head self-feed; this shortens the machine feed by the amount of the extension that will happen during the dwell time. ***Listing 7*** shows the G-code generated for a single tapping operation on a Tormach.

```
case "tapping":
      if (properties.reversingHead) {
        expandTappingPoint(x, y, z);
      } else {
      …
      }
      break;
    case "left-tapping":
      if (properties.reversingHead) {
        expandTappingPoint(x, y, z);
      } else {
      …
      }
      break;
    case "right-tapping":
      if (properties.reversingHead) {
        expandTappingPoint(x, y, z);
      } else {
      …
      }
      break;
```

Listing 5: Inside function `onCyclePoint()`, add three calls to the new function. Note that the entire existing content of each case goes inside the "else" braces. It wouldn't hurt to indent this content. Already done on Tormach.

Fusion 360 CAM for Tapping

The first CAM step is to define a new tool in the "Tap right hand" category. Assuming you already have the CAM to drill the hole, select a new drilling operation (or derived operation) on the same hole. Change the tool to the tap and verify your RPM. If you switch to the last panel of the Drill dialog, the cycle should have automatically changed to "Tapping," as shown in ***Figure 2***. This is what will kick in the code for the self-reversing tapping head.

Be sure to check the heights as discussed before. Retract height should be more than the full extension of the tapping head. I adjust the bottom height by checking "Drill Tip Through Bottom" and entering the extra depth needed to get full threads all the way, as shown in ***Figure 3***. This can be entered as an equation – for example, 6/32 for six more threads at 32 TPI.

After running the post, take a look at the G-code and see if it meets expectations. This is where I noticed the feed rate being rounded to one decimal place. It only got

```
} else {
    if (cycleExpanded) {
      expandCyclePoint(x, y, z);
    } else if (((cycleType == "tapping") || (cycleType == "right-tapping")
|| (cycleType == "left-tapping")) && properties.reversingHead) {
      expandTappingPoint(x, y, z);
    } else {
```

Listing 6: Near the end of `onCyclePoint()`. Already done on Tormach.

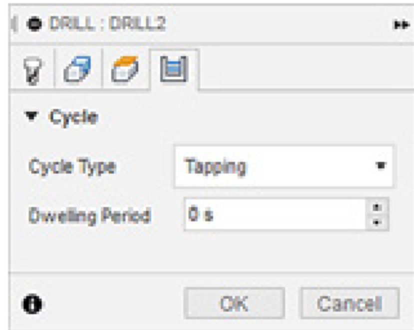


Figure 2: Cycle tab on the Fusion 360 drilling properties, showing "Tapping" is selected.

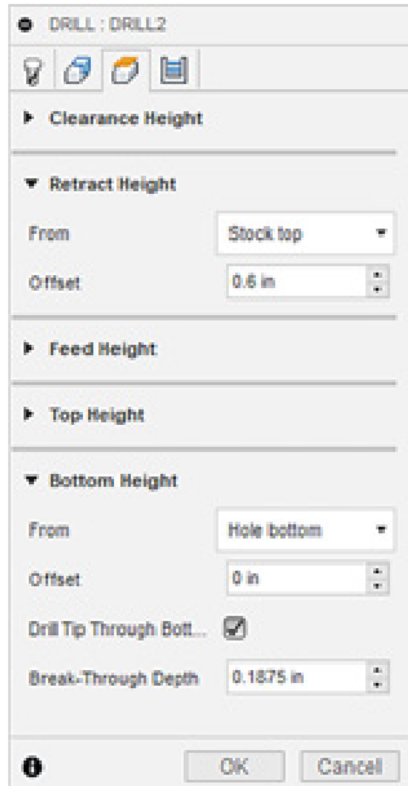


Figure 3: Heights tab on the Fusion 360 drilling properties, showing a little extra depth (6/32) to ensure full threads despite the tap chamfer.

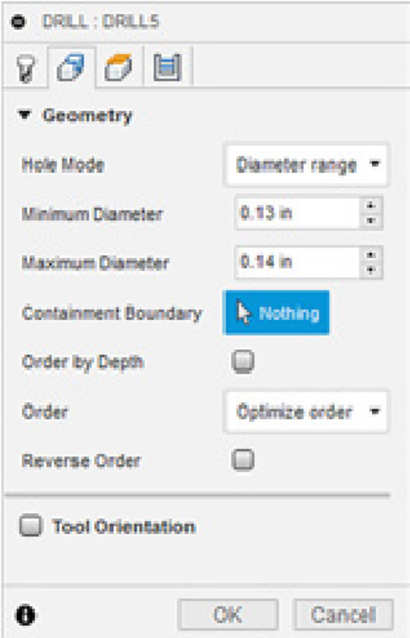


Figure 4: Geometry tab on the Fusion 360 drilling properties, showing a hole range that will automatically select 8-32 holes.

my attention because it rounded up and feeding too fast should be avoided. Feeding a little slow isn't a problem, as the feed will catch up before the tapping head has extended to neutral.

Once you have the recipe perfected and you've tested it out, lock it in. By that I mean save it as a Fusion 360 CAM template. But first, adjust both the drilling and tapping operations on the Geometry tab so the "Hole Mode" is diameter range. I use a range of .01". For example, an 8-32 hole is typically drilled at .136" (although the Fusion 360 Hole command makes it .1345"). So, I use a hole size range of .13" to .14", as shown in ***Figure 4***. Using the size range instead of picking the holes explicitly will allow the template to automatically select the right holes on other designs. Select the drill and tap operations at the same time, right click, and choose "Store as Template." A clear name like "8-32 threaded hole" will make it easy to find.

Once you have a template, it is simple to apply it. Just right click on a CAM setup, hover on "Create from Template," and select the template from the flyout. You need to manually tell Fusion 360 to generate tool paths (Ctrl-G), but you don't need to edit the drill and tap operations you just added. They'll automatically find the holes by size and use all the settings you saved. Happy tapping!

```
N10(Drill2)
T11 G43 H11 M6
S576 M3 M8
G54
G0 X0. Y0.
G0 Z0.7
G0 Z0.6
M49
G1 Z-0.2375 F18.
G4 P0.667
G1 Z0.6 F28.8
M48
G80
G0 Z0.7
M5 M9
```

Listing 7: Fusion 360 CAM output using the tapping cycle.