PathPilot™

Tormach's Control Architecture for the Future

This white paper provides a brief technical discussion of PathPilot™, the next generation motion controller for Tormach PCNC mills and the Tormach 15L Slant-PRO lathe. The release of PathPilot is the culmination of a multi-year development effort by our engineering team to create a new control architecture that addresses key shortcomings and obsolescence risks of our previous controller, while providing an improved operator experience, additional high-value functionality, and equal economy.

Engineers often fret over what they call control architecture. People outside the world of engineering wonder how it is that a common term like architecture could possibly be used to describe a bunch of wires, software, and electronics. PathPilot isn't a physical building, of course, but there are analogies.

Components like the trajectory planner and G-code interpreter are the foundation and structural elements — fundamental in keeping the building from collapsing, but rarely given a second thought as long as they work as designed.

Others, like the human machine interface and conversational programming utility are the street facing components like windows, doors, and siding. These design elements are visually obvious from a distance, but also must have underlying utility. A door to nowhere isn't particularly useful, no matter how nice it looks.

One thing that is often overlooked is what happens after the doors first open. The building needs a maintenance plan. To that end, we've invested internally to support PathPilot long term with dedicated technical and development staff, published bug lists, and a development schedule for future enhancements that will increase utility and the confidence of our customers in PathPilot and our products that use it.

We built PathPilot for our present-day product offerings, but also with an eye to the future. Sometimes we look at a building and it's painfully apparent that there have been additions, modifications, or revisions over the years. Other times the changes that occur become almost invisible — the building looks and feels complete and proper both before and after the modification. This ability to avoid obsolescence with a design that has flexibility for the future, is key aspect of proper architecture whether it's the architecture of a building or a computer. With PathPilot, I'm confident that we've achieved that. The result is a control architecture that can grow with our products. We're proud to offer PathPilot to existing and future Tormach customers.

Greg Jackson

President, Tormach Inc.

1. Why is PathPilot™ Needed?

Nearly 10 years has elapsed since we debuted our original CNC machines, the first generation of Tormach PCNC 1100 mills. Our ambition was to bring to market a uniquely affordable, accessible and capable CNC mill that could be used outside of the typical confines of factory production. To that end, PCNC mills have been enthusiastically adopted in education, small shops, by hobbyists, inventors, and entrepreneurs. Today, that goal and our related mission statement, *Enabling Your Ideas*, remain largely the same.

In the years since, we've made several major changes and scores of incremental changes to the PCNC 1100 design to improve performance, expand utility, and improve quality. Current operators, however, would still recognize and be able to use the original control software in early PCNC 1100 mills. That software was based on Mach3, a popular PC-based motion-control platform licensed from its developers, Artsoft.

Early on, PCNC mill adopters where encouraged to supply their own PC to run the control software application, but changes in operating software and computer hardware architecture made it increasingly difficult for customers to be successful with this approach. Over time, we were compelled to take more ownership in the controller operating system — and specific controller hardware — in an effort to provide a suitable motion-control solution to our customers. This ultimately resulted in the creation of the Tormach Machine Controller (TMC), which combined MachOS™ — a purpose built operating system we created based on the Windows XP embedded platform — with specific motherboards and other hardware components benchmarked by our engineering team.

This approach was successful in resolving the majority of motion-related control issues, but some nagging problems remained, and maintaining supply chain of key hardware components was an ongoing challenge. The scarcity of motherboards with IEEE-1284 (printer port) support resulted in numerous hardware revisions to the TMC as suitable motherboards and chipsets reach end-of-life.

When ArtSoft announced their intention to end support for Mach3 several years ago, it stimulated an in-depth review of requirements for the next generation of motion control for our PCNC family of products. Reliable motion control was paramount, but cataloging operator experiences from conversations with customers and technical support records revealed other shortcomings as well. For example:

- Customers often struggled with the non-machine related computer functions.
- The interface was cluttered and difficult to use.
- Certain workflows, like setting tool and work offsets, were confusing and esoteric.
- CAD/CAM programming was a significant barrier to using the machine.
- Implementation flaws of core control features like soft limits and overrides detracted from the overall operator experience and reduced machine confidence.

In light of these observations, the following sections of this paper explain some of the major engineering decisions that were made during the development of PathPilot.

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

2. PathPilot as an Appliance

If you dismantle your PC, TiVo®, or office telephone exchange you would find that they are all, in fact, general purpose computers with CPU, RAM, and a data-storage device. The product case, operator controls, and display interface differentiate them and transform each into an appliance – a dedicated device optimized for a specific purpose. This optimization makes the product easier to learn and more efficient in everyday use.

PathPilot adopts this appliance philosophy. All the displays and controls available to the operator are directly related to CNC machine control. In no place is general purpose computer operating system knowledge required or assumed. This offers two advantages:

- The machine operator is working throughout in a familiar context
- Changes and developments in the underlying software will not affect the operation of the machine tool. Consider how disruptive a change in your desktop computer operating system (i.e., Windows XP to Windows 8) can be on the reliant application programs.

When powered on, PathPilot displays the machine tool control screen; you can use it to get G-code programs from a memory stick or network; to setup the work offsets; to run your machining programs; and, when you are finished, the exit button shuts everything down. The operator will never need to interact with PathPilot's underlying operating system.

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

3. PathPilot as a Human-Machine Interface

The Challenge of Simplicity

Good interface design considers the cumulative costs of complexity. Low-value features subtract from the quality of the overall system by cluttering the interface and confusing the operator. PathPilot utilizes four methods to create a highly functional yet simple and streamlined interface: *implicit customization, persistent location, visual cues,* and *standardized workflow.*

Implicit Customization

One way of removing complexity is to remove explicit customization (i.e., operator configurable settings) in favor of implied operator choices. As a simple example of how implicit customization reduces complexity, let's contrast how PathPilot remembers the active unit mode (G20 or G21) with the previous controller.

The previous controller defaulted to G20 inch mode upon power up. If an operator wanted to run the controller in G21 metric mode, two options were available:

a) Explicitly set G21 each time after power up using interface buttons or a program line command, or

b) Permanently edit a configuration file outside of the interface and save it to memory.

PathPilot, on the other hand, remembers the units in use when the machine is powered off and restores them on start-up. All an operator has to do is run a G-code program written using G21 and the controller will remain in metric mode thereafter (at least until an inch program is executed). There is no need for a button to select metric or inch display. Similar implicit logic has helped simplify the operation of the automatic tool changer (ATC), probing, and work offset functions of the control without loss of functionality.

Persistent Location

Human factors' research has demonstrated that operators quickly learn the location of controls on a panel. Ensuring that a given function is always located in the same place on the screen greatly increases speed and minimizes errors in operating a machine.

To that end, we've made the primary displays and controls persistent, meaning that they are always shown at the same location on the bottom half of the screen (see **Figure 1**). A series of tabs allows different information to be shown on the upper half as requested by the operator. Functions displayed in the tabs are clearly organized into groups and never repeat in multiple tabs.

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

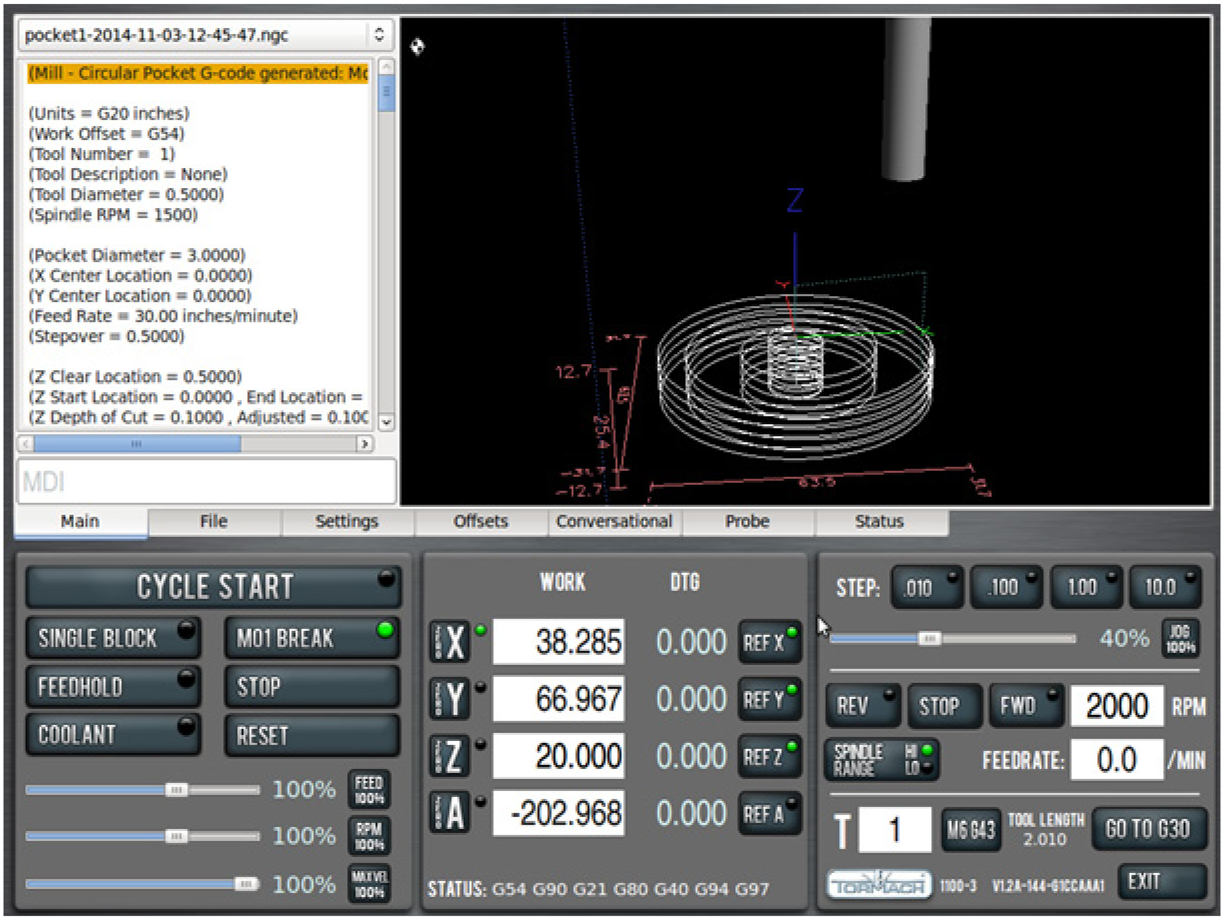

Figure 1

Visual Cues

The screen design of a computer interfaces is subject to current trends (see **Figure 2**). Both Windows Vista and Windows 7 had an elaborate semi-transparent, 3D look.

With the rapid advancement in touch and gesture recognition, newer operating systems such as Windows 8, Android, and Mac OS X have introduced flat, minimalist looks. While undoubtedly attractive, one key failing of this style is that it does not make it visually obvious where the operator should click on the screen.

PathPilot screens provide an impression of a physical machine control panel. Buttons appear 3D and *move* when clicked. The state of a button is clearly indicated by illuminated LEDs.

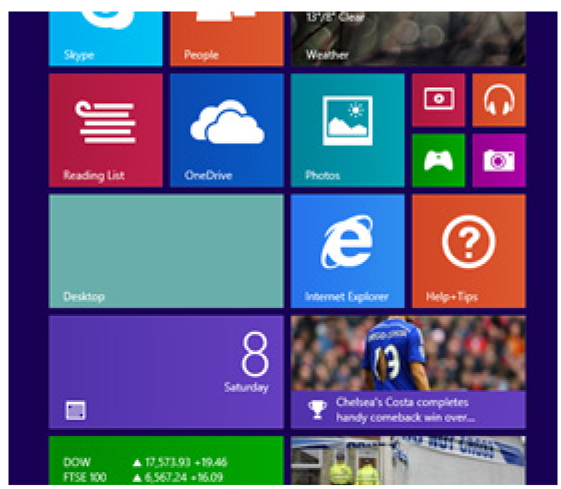

Figure 2

Numeric values are displayed in rectangular digital read outs (DRO) with contrasting background that turns blue when the DRO is selected for data entry. Simple sliders allow for override adjustments to feed rate or spindle speed.

This design style easily lends itself to a touch interface. PathPilot is designed to be touchscreen friendly. A virtual (soft) keyboard setting allows for the use of a touchscreen with or without a keyboard and mouse.

Standardized Workflow

A successful human-machine interface design features workflows that are natural and obvious to the operator. Our previous generation of control software fell short of that mark: features were overly complicated — both difficult to understand and cumbersome to use in practice. In hindsight, we tried too hard to explain how the CNC system actually works, when all operators really need to know is what to do to make accurate parts quickly.

Nowhere was this shortcoming more obvious than in the obtuse and confusing tool offset and work offset screens of the previous interface. Understanding tool offsets and work offsets has proven to be a steep part of every beginning CNC machinist's learning curve. There are many reasons for this, including:

- Terminology sounds abstract and mathematical to inexperienced operators.
- There are many different methods of setting up a job that all achieve equally good results.
- Different options purchased with the machine dictate the method offsets were applied.

PathPilot simplifies the offsets strategy by enforcing a standard workflow. In the new workflow the system uses the actual length of the tool cutting as measured from the underside of the machine spindle. For Tormach Tooling System (TTS) tools, this is the length from the TTS flange to the cutting tip.

Equally important, this strategy sets the same values into the system regardless of whether a TTS height gauge, electronic tool setter, simple feeler gauge or paper is used (see **Figure 3**). Since all methods are now treated identically, these can be used interchangeably and without confusion.

Figure 3

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

4. Functional Highlights

PathPilot has numerous new and improved features over the previous operating system. The following sections provides discussion on three key functional improvements that merit further discussion: *conversational programming, program overrides,* and *soft limits.*

Conversational Programming

Many simple milling tasks such as facing a piece of stock to a given thickness, cutting circular or rectangular holes in a panel, or drilling a bolt circle on a flange, do not require CAD/CAM programming. These tasks lend themselves to being quickly and easily programmed directly at the machine controller; this is commonly referred to as *conversational programming* or *shop floor programming.* PathPilot has an integrated on-board conversational programming interface to assist the operator in defining the geometry of standard machining features, choosing a suitable tool and its corresponding feeds and speeds, and creating a G-code program to machine it.

Programs generated with PathPilot's conversational programming can be saved and stored in the memory of the controller for future use. All programs are output with standardized formatting and commenting for programming consistency. Multi-operation programs can be easily created by chaining together unit operations to create a single G-code file.

The PathPilot conversational programming interface currently supports facing, profiling, pocketing, drilling, tapping, thread milling, and engraving (see **Figure 4**).

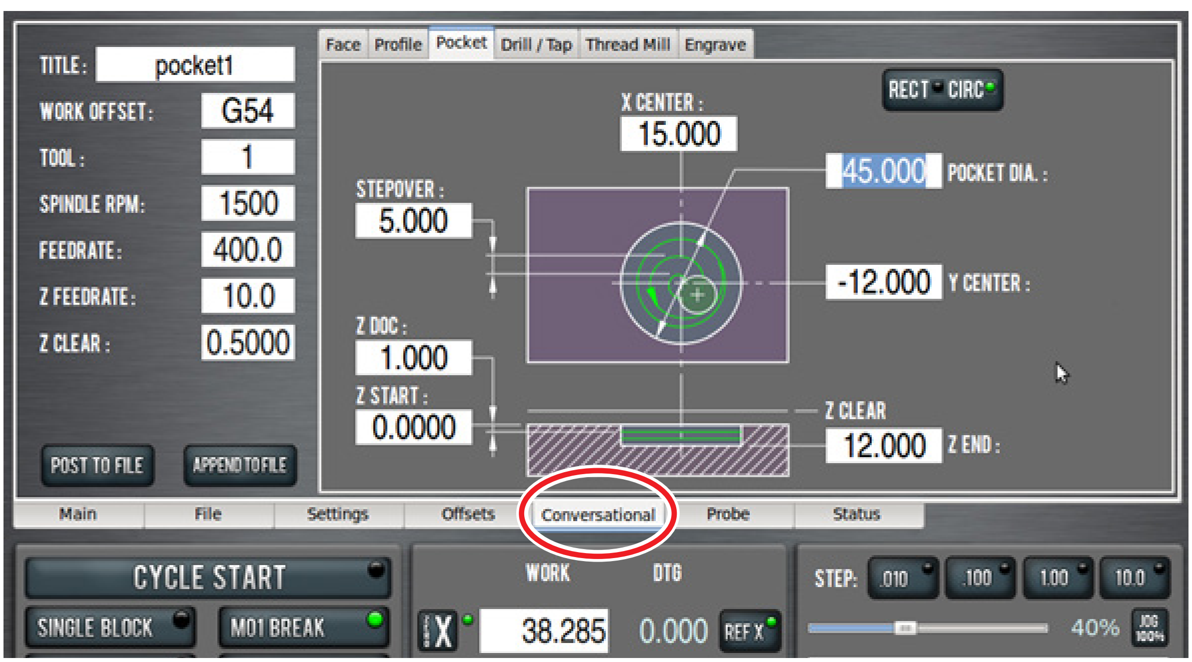

Figure 4

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

Program Overrides

Even the most experienced machinists cannot always predict optimal feed rates and spindle speeds prior to programming a part. Overrides on CNC controllers allow the operator to make real-time adjustments to the program based on what they see and hear, as the part is being cut.

The implementation of overrides in Mach3 had significant delays between operator action and machine response, which greatly limited their utility to the machinist. PathPilot eliminates these delays and increases operator confidence in the machine.

Operators can now confidently use overrides in a variety of workflows, including:

- Running a program in *slow motion* when using new G-code for first time to prove out a tool path.
- Adjusting feed rates and speeds to account for chatter, tool wear over time, etc.
- Temporarily pausing a program (*feed hold*) to clear chips or adjust the position of coolant stream.
- Stopping a program as necessary and quickly to address unplanned situations.

Soft Limits

Hitting a physical limit switch (i.e., a *hard* limit) was a common experience when using the original PCNC control system during the setup for large parts (when the operator was jogging near the edges of the work envelope of the machine). This was frustrating to operators as hitting a limit lost the reference state of all axes and required the machine to be referenced again by pressing the *Reset* button.

PathPilot addresses this by implementing software limits (i.e., *soft* limits). Running the machine past a soft limit results in a controlled stop prior to hitting any limit switch. Physical limit switches are retained, but used only for determining the location of the soft limits and not for directly stopping motion.

PathPilot's Soft limits have several advantages over physical limit switches, including:

- Soft limits are very repeatable (they exceed the repeatability of roller-contact limit switches).
- Should a limit switch become faulty, PathPilot provides accommodations so that soft limits can be manually set. In this case, the operator jogs by eye to the home position of each axis before clicking the reference button to set the location of the software limit.
- PathPilot is aware of soft limit locations. Any potential violations of soft limit rules trigger an error message when loading a G-code program. Also, the extents of the work envelope are accurately displayed in the tool-path window.

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534

5. Reliability of Motion

Tormach's stepper-based axis architecture design — used in the Series 3 PCNC 1100, PCNC 770, and the 15L Slant-PRO lathe — remains best-in-class in terms of precision, reliability, and economy. When presented with a stable command pulse stream, motion failures are virtually non-existent. Observed motion failures are almost always related to issues with the quality of the command-pulse stream generated by the controller. To ensure pulse-stream stability with the previous generation of controller, our engineer team implemented a number of strategies over the years. These include chipset benchmarking, purpose-dedicated Windows XP embedded Operating System (MachOS), and various electrical noise suppression strategies.

While MachOS outperformed standard Windows installations for pulse-timing stability, it was still occasionally vulnerable to specific lost motion trigger events. One such trigger event occurs when an operator plugs or unplugs a USB device while the machine is in motion. The ensuing Windows USB enumeration process took priority over motion-control pulsing and destabilized the pulse timing. This created a potential for a lost step or direction pulse. Similar triggers where associated with refreshing the tool-path display, computer networking applications, and electrically noisy shop environments.

Similarly, maintaining pulse-stream stability has become increasingly challenging with the exponential increases in power and complexity of personal computers. Chip designers have implemented a number of hardware and software techniques to reduce active heating in the chip. One commonly employed technique is to slow clock frequencies when intensive processing is not needed. Changes in clock frequencies cause problems when the chip is used to produce accurately timed pulses to drive motors, as was the case with our Mach3-based controller. It has been an ongoing challenge for us to mitigate the effects of these increasingly clever power saving techniques in CPU chips with respect to pulse timing.

PathPilot increases motion-control reliability by taking advantage of new hardware and software solutions to maintain pulse-stream quality in a wide array of usage scenarios. It uses dedicated hardware for pulse timing and firmware in a field programmable gate array (FPGA) with an operating frequency that can accept variations in processing speed that are 20 to 50 times greater than was possible before. This makes for a robust pulsing engine that will not fail in manners seen in the previous controller.

6. Conclusion

PathPilot is the next generation controller for Tormach products. It addresses a number of issues inherent in the previous control architecture and provides step advancements in reliability, utility, and workflow. PathPilot embraces the appliance concept to create a product that is simple to understand and use, and does not assume or require prior computing knowledge.

PathPilot also provides a foundation for future Tormach product development and feature enhancements to existing products.

Tormach Inc.
1071 Uniek Drive, Waunakee, WI 53597
Phone: 608.849.8381 / Fax: 209.885.4534