

CNC Machine Probing

CNC Machine Probe Simulation

Introduction

CNC Machine Probe Simulation enables you to simulate the use of probing devices on a CNC machining center and detect errors that could break the probe. The simulation feature can also be used to simulate any machine model stopping when it contacts another, such as moving a stock until it contacts a bar stop or moving steady-rest rollers until they contact the stock. Tool length setting and checking is not addressed directly at this time.

CNC Machine Probing is a licensed add-on module to Verification. You **DO NOT** need a probe license to create a probe tool in Tool Manager or to use a probe tool in VERICUT. You **DO** need a probe license to make the motion stop when the probe tip contacts something. Without a probe license, and without configuring the control for probe motion, you will get collision errors when the probe tool collides with something it should not, as described below.

Probe tool

A new probe tool type is added to **Tool Manager** for use when probing. The probe tip consists of a spherical "tip" of specified diameter, a shank/stem with a specified diameter, and an overall length. The probe tool can also have holder components used for collision detection.

A probe tool does not cut. In fact, VERICUT's material removal logic is never used for a probe tool. This is mainly because a probe tool, unlike a milling tool, does not have to be symmetric about an axis of rotation. Thus VERICUT can support multiple tip probe tool configurations like a "star" probes. This design also positions us to be able to support asymmetric probe tips in some future release.

VERICUT's collision logic for a probe tool is more like a turning tool. As with a turning tool, a probe tool uses machine simulation's collision algorithm to detect collisions. No material is removed, but errors are reported in the logger and log file, and the error counter is incremented. If a machine simulation view is open and simulation and collision checking is on, you will see the usual error color display.

Probe motion

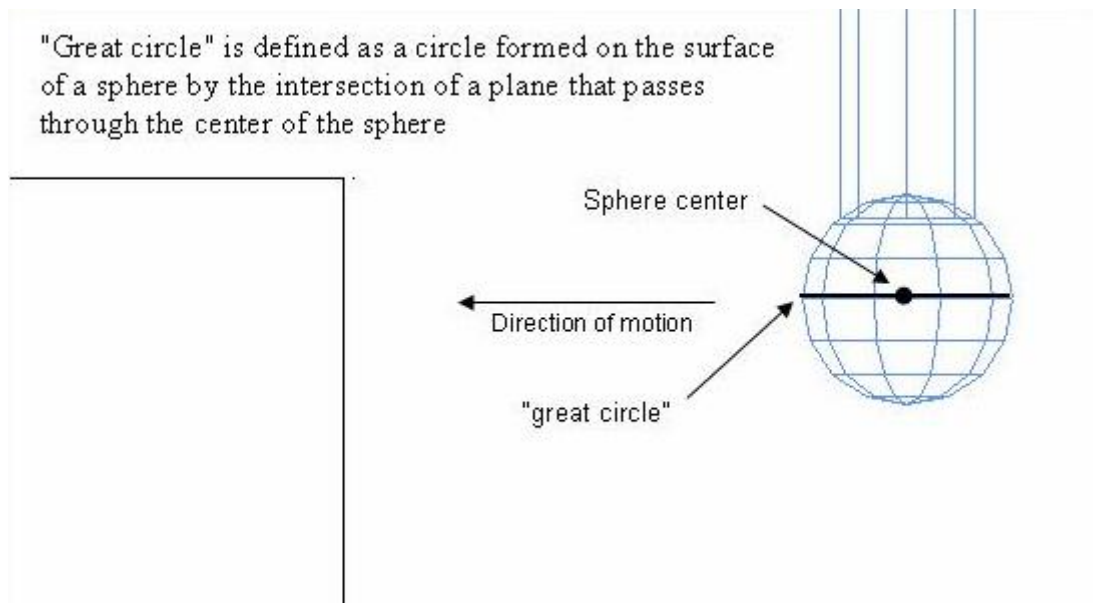
The G-Code macro **Probe** turns on probing for the current block. This macro is associated with the active tool component, and the corresponding tool must be a Probe Tool. Probe motion causes VERICUT to calculate the location where the probing component encounters an object between the start and the end point of the motion. The machine then moves to the probe contact location, or if no contact is made, to the end point of the motion.

NOTE: For additional information about **Probe**, and all VERICUT macros, see *VERICUT macros* in the *CGTech Help Library*.

Probing with a "Probe" Tool

When probing with a Probe Tool, the end of the motion is calculated as follows:

The spherical probe tip(s) are processed through a special algorithm that computes an accurate "hit" location. The location is based on the "great circle" of the sphere which lies in the motion plane.



The axes are moved to that location. If the spherical probe tip makes contact at any other location other than the "great circle", a warning message is output and motion stops at the alternate location. If no contact is detected, motion continues to the programmed location. Standard collision logic is used to determine if the probe tip hits any other component during the probe motion, but the motion continues to the probe's destination.

Stop Motion on Contact with Any Component

While not technically "probe" motion, this feature uses the same logic and can be used for moving a component, such as a steady rest roller, into contact with another component, such as the stock, or for moving the stock component into a bar stop. The G-Code macro **Touch** turns on "move until you touch" for the current block. The default "touch" component is the active tool. Use the G-Code macro **TouchComponentName** to specify a component other than the active tool.

NOTE: For additional information about these, and all VERICUT macros, see *VERICUT macros* in the *CGTech Help Library*.

When any portion of the touch component contacts any other component, the axes are moved to that location. The axes move until the touch component arrives at the programmed location, or until the object contacts something. Standard collision logic is used to determine if any other component hits something during the "touch" motion, but the motion continues to the touch component's destination.

When probing with something other than a Probe tool-type, VERICUT approximates the contact location. When any portion of the touch component contacts any other component, the axes are moved to that location and stop.

Collision Detection during Probe Motion

Machine Simulation and collision checking do not need to be "on" to detect collisions. As with turning tool holders, if the "Display Holders in Workpiece View" box is checked "on", then collisions between the holder and "workpiece visible" components are detected using the same algorithm used by machine simulation. A probe "holder" error states that a collision occurred between the "Probe Body" and workpiece components (typically Stock or Fixture).

During "Probe" motion, collision checking only looks at the "stem" portion of the Probe Tip, treating it as part of the holder. There is no collision checking applied to the spherical portion of the probe tip since motion stops when it contacts anything. A collision between the Probe Tip and any "workpiece visible" component generates an error stating that the "Probe Tip" and the workpiece components collided.

If a Probe tool is used during normal (not Probe) motion, collision checking also looks at the spherical Probe Tip and reports any collisions.

Position Feedback and Logic

NC program probe logic often uses feedback of axis positions when the probe contacts. VERICUT supports this using the current ability to assign axis positions to variables.

Notes about Probing in VERICUT

The following key points should be considered about Probe Tools and "workpiece view" collision checking (ie no machine simulation).

- A probe tool does not cut. In fact, VERICUT's material removal logic is never used for a probe tool. This is mainly because a probe tool, unlike a milling tool, does not have to be symmetric about an axis of rotation. Thus VERICUT can support probe tool configurations like a "star" probe. This design also positions us to be able to support asymmetric probe tips in a future release.
- VERICUT's collision logic for a probe tool is more like a turning tool. As with a turning tool, a probe tool uses machine simulation's collision algorithm to detect collisions. No material is removed, but errors are reported in the logger and log file, and the error counter is incremented. If a machine simulation view is open and simulation and collision checking is on, you will see the usual error color display.
- Machine Simulation and collision checking do not need to be on to detect collisions. As with turning tool holders, if the "Display Holders in Workpiece View" box is checked "on", then collisions between the holder and "workpiece visible" components are detected using the same algorithm used by machine simulation. A probe "holder" error states that a collision occurred between the "Probe Body" and workpiece components (typically Stock or Fixture).
- During "Probe" motion, collision checking only looks at the "stem" portion of the Probe Tip, treating it as part of the holder. There is no collision checking applied to the spherical portion of the probe tip since motion stops when it contacts anything. Collisions between the Probe Tip and any "workpiece visible" component generates an error stating that the "Probe Tip" and the workpiece components collided.
- During normal (not Probe) motion, collision checking also looks at the spherical Probe Tip and reports any collisions.
- VERICUT detects if machine simulation and collision checking is on and collisions between the Tool and any "workpiece visible" components is selected. Thus it does not report the collision twice. This occurs for turning tools and probe tools.
- You do not need a probe license to create a probe tool in Tool Manager or to use a probe tool in VERICUT. You do need a probe license to make the motion stop when the probe tip contacts something. Without a probe license, and without configuring the control for probe motion, you will get collision errors when the probe tool collides with something it should not, as described above.
- The following G-Code macros are used to control probing features in VERICUT:
Probe, **Touch** and **TouchComponentName**.