

Expressions:

Expressions follow clause references as described for G code 0 in the Supported G-Code document. Each expression must evaluate to a number at execution time or an execution error will occur. Numeric scalar literals are evaluated at compile time. Variable references are evaluated at execution time. More complex expressions including those with arithmetic operands and function calls are enclosed in square brackets and evaluated at run time.

Complex expressions are not syntax checked at compile time but this could change. Generally an expression can be substituted wherever a scalar value is valid. Expressions consist of a sequence of fixed floating point numbers, variable references, arithmetic operands and function calls. Parentheses can be used to alter sequence of evaluation.

Variables:

Variables resolve to a fixed floating point number and can generally be used in a program where a fixed floating point number is valid. Variable names are strings constrained to an alphabetic, upper and lower case, character set and the underscore character. A variable is identified by a leading pound symbol or by a leading underscore. A leading pound symbol indicates a temporary variable associated with the CNC document. A leading underscore indicates a persistent machine variable, assuming that the machine document is saved before exiting. CNC program execution may span two or more document files and machine variables may be referenced by any documents logically connected to the machine.

Arithmetic operands in order of precedence

Symbol	Description
<i>E</i>	<i>a * 10 ** b (base 10 exponent)</i>
<i>^</i>	<i>a ** b (value raised to a power)</i>
<i>%, +, /</i>	<i>modulo, multiply and divide</i>
<i>+, -</i>	<i>add and subtract</i>
<i>,</i>	<i>a comma or space will separate array values</i>

Functions (case insensitive)

Name	Description
<i>TAN(a)</i>	<i>Trigonometric TANGENT, units for a is degrees, result ranges between -1 and 1 inclusive.</i>
<i>ARCTAN(x)</i>	<i>Inverse TANGENT, $-1 \leq x \leq 1$, result ranges from -180° to 180° inclusive.</i>
<i>COTAN(a)</i>	<i>1 / TANGENT(a)</i>
<i>ATAN2(y, x)</i>	<i>Inverse TANGENT but with result ranging from 0° (inclusive) to 360° (exclusive).</i>
<i>LN(x)</i>	<i>Natural Logarithm</i>
<i>LOG(x)</i>	<i>Common Logarithm (base 10)</i>
<i>LOG2(x)</i>	<i>Logarithm base 2</i>
<i>EXP(x)</i>	<i>Exponent function (equivalent to "^" operator)</i>
<i>SQRT(x)</i>	<i>Square root</i>
<i>ABS(x)</i>	<i>Absolute value</i>

Special read-only variables (case insensitive)

Name	Returns
<i>PI</i>	<i>3.141592..</i>
<i>__inc(remental)</i>	<i>0 if in absolute mode, 1 if in incremental mode</i>
<i>__abs(olute)</i>	<i>1 if in absolute mode, 0 if in incremental mode</i>
<i>__feedrate</i>	<i>active feed-rate specification</i>
<i>__(optional_)stops</i>	<i>1 if optional stops enabled, 0 not active</i>
<i>__(circular_)plane</i>	<i>selected contour plane, 0 if XY, 1 if ZX, 2 if YZ</i>

Name	Returns
<code>__tool_diameter</code>	<i>selected tool diameter or zero if no selected tool</i>
<code>__tool</code>	<i>selected tool number or 0 if none</i>
<code>__fixture</code>	<i>selected fixture number or 0 if none</i>
<code>__pos(ition)(_of)</code> <code>(_axis)_AXIS</code>	<i>current position of AXIS (X,Y,Z,A,B or C)</i>
<code>__probe_stop</code>	<i>return code from last probing move</i> <i>0 = normal exit</i> <i>-1 = probe already active at start of move</i> <i>1 = no probe contact</i>
<code>__analog(ue)(input_)</code> <code>(pin_)(PORT PIN)</code>	<i>PORT PIN is (AIB)(1-25); return range is 0 to 1 for input range 0 to 3.3 volts</i>
<code>__digital_(input_)</code> <code>(pin_)(PORT PIN)</code>	<i>PORT PIN is (AIB)(1-25); returns 1 if active or 0 if inactive; active high or active low depends upon active-high mode configuration.</i>
<code>__PARAMETER</code>	<i>PARAMETER is any single letter that may be set by a subroutine call</i>

Program Control

The RS274-D standard includes M-codes for jump, call and return but not for conditional execution. The execution of any line of code can be made conditional by adding the following syntax to the line:

if {value, variable or expression} {relational operator} {value, variable or expression}

Note that this syntax can be added at either the start or end of the line.

Relational Operators (case insensitive)

Symbol	Description
<i>LT or <</i>	<i>Less than</i>
<i>LE or <=</i>	<i>Less than or equal</i>
<i>EQ or ==</i>	<i>Equal to</i>
<i>NE or !=</i>	<i>Not equal to</i>
<i>GT or ></i>	<i>Greater than</i>
<i>GE or >=</i>	<i>Greater than or equal to</i>

Commands:

As an intermediate file format specification, the RS274-D standard is cryptic using single letter codes for commands and parameters. Although machinists will memorize codes, the use of string substitutions improve the readability of programs.

Parameter Substitutions (case insensitive)

Code	String substitution
<i>D</i>	<i>dia(meter)</i>
<i>F</i>	<i>(set)feedrate(to)</i> <i>Note: This parameter is modal and can be executed either from a separate line or on a movement command line.</i>
<i>I</i>	<i>centerX (circular interpolation)</i>
<i>J</i>	<i>centerY (circular interpolation)</i>
<i>K</i>	<i>centerZ (circular interpolation)</i>
<i>L</i>	<i>max(imum) segment angle</i> <i>(circular to cubic Bezier conversion)</i>
<i>P</i>	<i>time or duration in seconds</i>
<i>Q</i>	<i>peck or delta (drill cycle)</i>

Code	String substitution
<i>R</i>	(arc)radius (<i>circular interpolation</i>) or release(plane) (<i>drill cycle</i>) or angle (<i>rotate object coordinates</i>)
<i>S</i>	<i>spindle speed (RPM)</i>
<i>T</i>	(<i>select</i>)tool(<i>number</i>) <i>Note: This parameter is modal and can be executed either from a separate line or on a movement command line. Tool 0 is no tool.</i>