



Basic APT commands

\$\$	Comment
CIRCLE / cx,cy,cz, nx,ny,nz, radius	If a GOTO is preceded by a CIRCLE that GOTO represents the end point of the arc. The CIRCLE command specifies the center, the plane direction and the radius. GOTO / startx, starty, startz, vx,vy,vz CIRCLE / cx,cy,cz, nx,ny,nz, radius GOTO / endx, endy, endz, vx,vy,vz The direction of the arc is decided by the direction of the plane normal Ex. GOTO / 100,0,0,0,0,1 CIRCLE / 0,0,0,0,0,1,100 GOTO / 0,100,0,0,0,1
COOLNT / ON COOLNT / OFF COOLNT / mode	Enables/Disables the coolant Modes: MIST, FLOOD, TAPKUL, THRU, TAP The MIST mode is usually used for finishing operations. The FLOOD mode is used to remove a lot of material. The TAPKUL and TAP modes are used for tapping operations. The THRU mode is used to inject the cooling liquid inside the tool.
CUTCOM/ LEFT CUTCOM/ RIGHT CUTCOM/OFF	It enables or disables the tool radius compensation. LEFT and RIGHT are used to specify the side of the compensation
END	Program end
FEDRAT / mm_x_minute, MMPM FEDRAT / mm_x_second, MMPS FEDRAT / inch_x_minute, IPM	Feed rate

GOHOME	Moves at home position
GOTO / x, y, z, x component of the tool direction, y component of the tool direction, z component of the tool direction	<p>linear movement.</p> <p>If the GOTO is preceded by a RAPID the movement will be executed at rapid otherwise it will be executed at current feed rate.</p> <p>Example: RAPID GOTO / 100,200,300,0,0,1 GOTO / 1000,200,0,0,0,1</p> <p>The first GOTO is at rapid while the second is at feed.</p>
INSERT / any text string	Inserts a text string directly on the output file.
LOADTL / n LOADTL / n, IN , pos LOADTL /n, OSETNO , corrector	<p>Load the tool n. By default the tool is loaded into the kinematics tool node having index 1. The optional IN parameter (requires at least Eureka 4.9.2260) allows to select a different kinematics node.</p> <p>The parameter OSETNO specifies the corrector used for radius and length tool compensation (valid only for machine tool)</p>
RAPID	Set following GOTO movement with RAPID feed.
SPINDL / RPM , round per minute, CLW SPINDL / RPM , round x minute, CCLW SPINDL / ON SPINDL / OFF	<p>Spindle control.</p> <p>By default the command addresses the first spindle available in the kinematics tree. All SPINDL commands can have an optional MOTOR parameter (requires at least Eureka 4.9.2260). If there are more spindles, the parameter MOTOR allows to select the correct spindle.</p> <p>For example: SPINDL / ON, MOTOR, 2</p>

Robot extensions

<p>BREAK</p>	<p>Calls the OnPostProcBreak event. It can be used to split the output file for example.</p>
<p>CALSUB / subname CALSUB / subname, ARGS, par1_name, par1_value, par2_name, par2_value</p>	<p>Calls a subprogram. The subprogram has to be declared in the subprogram list. Optionally it is possible to pass an unlimited number of arguments specifying a set of couples parameter name,parameter value. In the called subprogram arguments can be used inside commands using placeholders {arg1}, {arg2} etc. Example: CALSUB / MYPROC, ARGS, F, 1000,X,100 In MYPROC it is possible to have: FEFRAT / {arg1},MMPM MOVETO / X,{arg2}</p>
<p>CSYS / object frame index</p>	<p>Select an object frame</p>
<p>CUTTER / d, r, a, b, alpha, beta, h CUTTER / d, r, a, b, alpha, beta, h, TYPE, ADDITIVE CUTTER / d, r, a, b, alpha, beta, h, TYPE, ADDITIVE, TRACECOLOR, index</p>	<p>Defines basic tool geometry adding this commands after the LOADTL command. The tool defined from APT will appear in the tool database but they are considered as “temporary” tools. All temporary tools are removed when a simulation is restarted. d = cutter diameter r = corner radius alpha = tip angle beta = edge angle h = cutter height Add TYPE, ADDITIVE for additive tool. Add TRACECOLOR index (0 to 19) to set the tool trace color.</p>
<p>CUTTER / SHANK, d, 0, 0, 0, 0, 0, h CUTTER / SHANK, d, 0, 0, 0, 0, 0, h, TYPE, ADDITIVE, RGBA, r,g,b,a</p>	<p>Follows the previous command and add a cylinder shaped shank. d is the diameter and h is the cylinder height. The tool length will be the sum of the shank height and the cutter height. Add TYPE, ADDITIVE, RGBA to define shank color and transparency on additive tool. r = red scale value (0 to 1) b = blue scale value (0 to 1) g = green scale value (0 to 1) a = transparency scale value (0 to 1)</p>

<p>GOTO / x, y, z, x component of the tool direction, y component of the tool direction, z component of the tool direction, EXT, e1, ...</p>	<p>Optionally the GOTO command can force the position of the external axes using the EXT parameter followed by the external axes values</p>
<p>GOTO / ROTZ, rotangle, x,y,z,vx,vy,vz, EXT, e1, ...</p>	<p>The ROTZ parameter adds a rotation around the tool axis to the one chosen by Eureka. External axes can be specified to force their position (requires at least Eureka 4.9.2260).</p>
<p>GOTO / x,y,z,vx,vy,vz, VREF, wx,wy,wz,angle</p>	<p>The VREF parameter add a constrain in order to align the X axis of the target frame to a given direction. The wx,wy,wz components decide the reference direction. The angle value (in degrees) allows to rotate the direction around the Z axis.</p>
<p>MODE / INTERP, ON, TYPE, n MODE / INTERP, OFF, TYPE, n</p>	<p>Set the automatic optimization strategy for external axes. Use Eureka Robot External axis interpolation configuration window to define it.</p>
<p>MOVEJ / j1, j2, j3, j4, j5, j6</p>	<p>Joint-interpolated, point to point movement where the target joint position and the target external axes position is given.</p>
<p>MOVEJ / j1, j2, j3, j4, j5, j6, EXT, e1, ...</p>	<p>Optionally the MOVEJ command can force the position of the external axes using the EXT parameter followed by the external axes values.</p>
<p>MOVEJ / x,y,z, RPY, rz, ry, rx, CFG, 0,0,0,0, EXT, e1, ...</p>	<p>Joint-interpolated, point to point movement where the target robot position is specified as cartesian position plus a configuration that encodes the inverse-kinematics solution and the target external axes position is given. External axes are given as well. The orientation angles are in degrees and they can be specified in different ways: RPY: roll pitch yaw XYZ: rx*ry*rz ZYZ: rz*ry*rz QUAT: quaternion</p>
<p>MOVEL / x, y, z, RPY, rz, ry, rx, EXT, e1,...</p>	<p>Linear cartesian movement. External axes can be specified to force their position (requires at least Eureka 4.9.2260). The orientation angles are in degrees and they can be specified in different ways: RPY: roll pitch yaw XYZ: rx*ry*rz ZYZ: rz*ry*rz QUAT: quaternion</p>

MOVEL / x, y, z, RPY , rz, ry, rx, EXT , e1, REL	The REL parameters transform the MOVEL in a relative movement. The target is referred to the starting position/orientation.
MOVEL / x, y, z, VZVX , vz_i, vz_j, vz_k, vx_i, vx_j, vx_k, EXT , e1	With this syntax, the MOVEL constrains the z-axis direction and the x-axis direction of the tool frame along the correspondent direction specified using the direction cosines.
MOVEL / x, y, z, VZVY , vz_i, vz_j, vz_k, vy_i, vy_j, vy_k, EXT , e1	With this syntax, the MOVEL constrains the z-axis direction and the y-axis direction of the tool frame along the correspondent direction specified using the direction cosines.
MOVEL / x, y, z, VZNVX , vz_i, vz_j, vz_k, vnx_i, vnx_j, vnx_k, EXT , e1	With this syntax, the MOVEL constrains the z-axis direction and the negative x-axis direction of the tool frame along the correspondent direction specified using the direction cosines.
MOVEL / x, y, z, VZNVY , vz_i, vz_j, vz_k, vny_i, vny_j, vny_k, EXT , e1	With this syntax, the MOVEL constrains the z-axis direction and the y-axis direction of the tool frame along the correspondent direction specified using the direction cosines.
MOVETO / axisname, axispos	Joint-interpolated, point to point movement of a specified set of axes. For example the command MOVETO / A, 100, B, 200 moves only axes A and B
MOVETO / axisname, axispos, REL	The REL parameters transform the MOVETO in a relative movement. The target is referred to the starting position.
PPRINT / NC SEQUENCE NAME : sequence_name	Starts a new NC sequence and calls OnPostProcPPrint event.
PROBE /HOLE , x,y,z, DIAM , d, NPOINTS , n, STARTANG , a	Simulates a probe cycle that touches a certain number of points to get a hole position. Then it calls the OnPostProcProbe event.
PROBE /POINT ,x,y,z	Simulates a probe cycle that touches a point and calls the OnPostProcProbe event.
SEQUENCE / BEGIN , sequence name SEQUENCE / END	Begins/ends a new sequence. sequence_name: define name of sequences
SET / FLOW , flow_value, IN ,nozzle_index	Set the flow value for a given nozzle
SET / PAR , name, VALUE	Sets an APT parameter. APT parameters can be retrieved using the API GetAptPar of the controller object.

<p>SET / SIGNAL, name, VALUE, value</p>	<p>Sets digital or analogue output variables. For example: SET / SIGNAL, POWER, VALUE, 100</p>
<p>STRAT / strategy</p>	<p>Set the strategy for the GOTO processing and define complete orientation of TOOLFRAME. Use Eureka Robot GOTO Strategy configuration window to define it.</p>
<p>TLCOR / ON, x,y,z TLCOR / ON, x,y,z, RPY, rz,ry,rx TLCOR / ON, x,y,z, XYZ, rz,ry,rx TLCOR / ON, x,y,z, ZYZ, rz,ry,rx TLCOR / ON, x,y,z, QUAT, q1,q2,q2,q4 TLCOR / OFF</p>	<p>The behavior of TLCOR depends on the controller parameter TLCOR_ACTION. If its value is PRESET_TOOLFRAME, the effect of TLCOR is to adjust the preset of the toolframe with an additional transformation. This effect takes place if the TOOL_PRESET parameter is greater than 0. If the value is TRANSFORM_COORD, Eureka will add the TLCOR transformation to the target found automatically after a GOTO or MOVEL. Use TLCOR/OFF to disable the TLCOR effect.</p>
<p>TLCOR/ TYPE,ATTRACTIVE_POINT, x, y, z, CSYS, object_frame_index TLCOR/ TYPE,REPULSIVE_POINT, x, y, z, CSYS, object_frame_index</p>	<p>Eureka will add the TLCOR transformation to the target to have the z axes of the TOOLFRAME pointing towards or against a point in a selected object frame.</p>
<p>TLCOR/ TYPE, ATTRACTIVE_AXIS, x, y, z, i, j ,k, CSYS, object_frame_index TLCOR/ TYPE, REPULSIVE_AXIS, x, y, z, i, j ,k, CSYS, object_frame_index</p>	<p>Eureka will add the TLCOR transformation to the target to have the z axes of the TOOLFRAME pointing towards or against an axis in a selected object frame.</p>
<p>TLCOR/ ZDIR,i, j, k, CSYS, object_frame_index</p>	<p>Eureka will add the TLCOR transformation to the target to force the z axes of the TOOLFRAME always in a fixed orientation in a selected object frame.</p>
<p>TORCH / ON TORCH / ON, PROFILE, profile TORCH / OFF</p>	<p>Turns on/off the torch and calls the OnPostProcTorch event. The command can take additional parameters as couple name, value. For example, : TORCH / ON, PROFILE, 1 The parameter value can be any text string: TORCH / ON, PROFILE, R[14]</p>
<p>TRANS / x,y,z TRANS / x,y,z, RPY, rz,ry,rx TRANS / x,y,z, XYZ, rx,ry,rz</p>	<p>Transforms the current coordinate system. This command modifies the current coordinate system without affecting the original object frame.</p>

<p>TRANS / x,y,z, ZYZ, rz,ry,rz TRANS / x,y,z, QUAT, q1,q2,q2,q4</p>	<p>For example: TRANS / 100,200,300 translates the current coordinate system at x=100, y=200, z=300 with respect to the object frame. This command can be used for example to replicate a toolpath in several position or to offset a toolpath to make some kind of adjustment.</p> <p>Use TRANS /0,0,0 to reset its effect.</p>
<p>UNITS / MM UNITS / INCHES</p>	<p>Selects the programming unit</p>
<p>WAVELING / ON WAVELING / ON, PROFILE, profile WAVELING / OFF</p>	<p>Turns on/off the waveling and calls the OnPostProcWaveling event. The command can take additional parameters as couple name, value. For example, : WAVELING / ON, PROFILE, 1 The parameter value can be any text string: WAVELING / ON, PROFILE, R[14]</p>