TurboCAD Parametric Parts (PPM) — 1-Page Cheat Sheet

Purpose: Quick reference for writing TurboCAD Parametric Part Model (PPM) scripts. Print/save this page as PDF.

Edition Note: Some operations depend on TurboCAD edition/version. Use examples bundled with TurboCAD and the Help/Reference manual for exact signatures.

---

1) Script Skeleton

```
// Description: <what this PPM does>
// Author: <name or Unknown>
// Date: <YYYY-MM-DD>
// Notes: Units, dependencies (relative paths), version

Units(1[in]);                    // set working units

Input(a, b, c);                  // optional: for callable macros

P1 = Parameter("Name", 10, LINEAR, GreaterThan(0));
A1 = Parameter("Angle", 30, ANGULAR);
C1 = Parameter("Color", 0xff, COLOR);

// ... geometry ...

Output(result);                  // return 1+ results
```

Common Parameter Types

`LINEAR`, `ANGULAR`, `COLOR`, `MATERIAL`, `TEXT`, `FONT`, `CHECKBOX`

Constraints Helpers

`Interval(min,max)`, `Set(v1,v2,...)`, `GreaterThan(x)`, `GreaterOrEqual(x)`, `LessThan(x)`, `LessOrEqual(x)`

---

2) 2D Geometry

```
// Points & helpers
P  = Point(x, y);

// Polyline with lines, arcs, fillets (closed or open)
poly = Polyline(
  Point(0,0),
  Point(W,0),
  Arc1(cx, cy, r),   // CCW arc segment joining prior/next vertices
  Fillet(r),         // fillet at the corner being created
  Point(0,H),
  Point(0,0)
);

// Circle (center at origin unless moved)
cr = Circle(R);
```

Tips - Close a profile by repeating the start point as the last vertex in `Polyline(...)`. - `Arc1(cx,cy,r)` creates a circular arc segment between surrounding vertices.

---

3) Build 3D From 2D

```
// Extrude a closed 2D profile
solid = Extrude(profile2D, Height);

// Create a prismatic “tube” from a 2D circle (example pattern)
tube  = Thickness(Circle(R), L);   // L along +Z
```

4) 3D Primitives

```
box    = Box(0,0,0,  X, Y, Z);
sphere = Sphere(R);
```

5) Booleans

```
u = BooleanUnion(a, b);
s = BooleanSubtract(a, b);
i = BooleanIntersect(a, b);
```

6) Transforms

```
m = Move(obj, dx, dy, dz);
r = RotateZ(obj, angDeg, ox, oy); // rotate in XY about (ox,oy)
// (RotateX/RotateY/Scale variants may be available per version)
```

7) Text & Properties

```
// Place text at origin; use TextStyle to control anchor
s0 = Text(TXT, TextFont(0, 1, 0, FONTNAME), TextStyle(CENTER, MIDDLE, ITALIC));
s1 = SetProperties(s0, "PenColor" = CLR);    // apply color/material/layer, etc.

// Move to location (x0,y0)
s2 = Move(s1, x0, y0, 0);
```

Anchors: `LEFT | CENTER | RIGHT` and `TOP | MIDDLE | BASELINE | BOTTOM` (in `TextStyle`).

---

8) Utilities & Points

```
RP = RefPoint(x, y, z);                // set insertion handle
PP = ParameterPoint(x, y, z);          // user-pickable point
x0 = PointX(PP);  y0 = PointY(PP);     // extract coordinates
```

---

9) Patterns You'll Reuse

A) Closed Rectangle (Polyline)

```
rect2D = Polyline(
  Point(0,0), Point(W,0), Point(W,H), Point(0,H), Point(0,0)
);
```

B) Obround (slot) Profile

```
R = W/2;
slot = Polyline(
  Point(R,0), Point(L-R,0), Arc1(L-R, R, R),
```

```
  Point(L-R,W), Point(R,W), Arc1(R, R, R), Point(R,0)
);
```

C) Box with Through-Hole (boolean pattern)

```
box = Box(0,0,0, X, Y, Z);
tube = Thickness(Circle(HoleR), Z + 2*HoleR);
tube = Move(tube, X/2, Y/2, -HoleR);
result = BooleanSubtract(box, tube);
Output(result);
```

---

10) Best Practices

- **Header block**: description, author, date, units, dependencies (relative paths).
- **Consistent names**: keep `Input(...)` names identical to `Parameter(...)` names so callers can override defaults cleanly.
- **Clamp parameters** with constraints (`GreaterThan(0)`, `Interval(...)`).
- **Return minimal results**: output only what you need (group if appropriate).
- **Macros**: put shared functions in a `Macro` subfolder and call via **relative paths** per your standards.

---

11) Mini Templates

2D Closed Sketch Template

```
Units(1[in]);
Input(L, W);
L = Parameter("Length", 6, LINEAR, GreaterThan(0));
W = Parameter("Width",  2, LINEAR, GreaterThan(0));
R = W/2;
profile = Polyline( Point(R,0), Point(L-R,0), Arc1(L-R,R,R),
                    Point(L-R,W), Point(R,W), Arc1(R,R,R), Point(R,0) );
Output(profile);
```

3D From Sketch Template

```
Units(1[in]);
Input(W,H,Z);
W = Parameter("Rect Width", 4, LINEAR, GreaterThan(0));
H = Parameter("Rect Height",3, LINEAR, GreaterThan(0));
Z = Parameter("Extrude",    1, LINEAR, GreaterThan(0));
```

```
rect2D = Polyline(Point(0,0),Point(W,0),Point(W,H),Point(0,H),Point(0,0));
solid  = Extrude(rect2D, Z);
Output(solid);
```

Resources - TurboCAD Help ▶ Reference Manual (Parametric Parts) - IMSI Design Documentation Portal (imsidesign.com) - TurboCAD User Forum (community examples & tips)