

BSPPFit Model API Reference

v0.145.0.349

Peter Felecan assisted by Claude (Anthropic), model Sonnet 5

LibFile: bsppfit.scad

Parametric BSPP (British Standard Pipe Parallel) pipe fittings, built from a single port-list super-class and a catalogue of named wrapper modules --- pipe, elbow, tee, cross, reducer, manifold, cap, plug and nipple --- plus hex-bush, hose-tail and union decorations.

To use, add the following lines to the beginning of your file:

```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
```

File Contents

1. Section: Size and Dimension Tables

- `bsppfit_lookup()` – The dimension-table row for a BSPP size.
- `bsppfit_designator()` – The "G" thread designator for a BSPP size.
- `bsppfit_major_od()` – The thread major (crest) diameter for a BSPP size.
- `bsppfit_minor_od()` – The thread minor (root) diameter for a BSPP size.
- `bsppfit_pitch()` – The thread pitch for a BSPP size.
- `bsppfit_useful_thread_length()` – The useful thread engagement length for a BSPP size.
- `bsppfit_safe_wall()` – A requested outer diameter clamped to a safe wall thickness.
- `bsppfit_default_bore()` – The default through-bore diameter for a BSPP size.
- `bsppfit_safe_middle_od()` – A requested middle-band diameter clamped to a safe wall.
- `bsppfit_sizes()` – The list of supported nominal BSPP sizes.
- `bsppfit_designators()` – The list of supported "G" thread designators.

2. Section: Diagnostics and Utilities

- `bsppfit_cutaway()` – Section the child geometry open on one side, for inspection. [Geom]

3. Section: Port Representation

- `bsppfit_arm_od()` – The default arm outer diameter for a BSPP size.

4. Section: The Fitting Super-class

- `bsppfit()` – The fitting super-class --- builds any BSPP fitting from a port list. [Geom]

5. Section: Port-List Builders

- `bsppfit_pipe_ports()` – Port list for a straight pipe --- two coaxial same-size ports.
- `bsppfit_cap_ports()` – Port list for a cap's threaded socket --- one open female port.
- `bsppfit_tee_ports()` – Port list for a tee --- a run plus one branch.
- `bsppfit_cross_ports()` – Port list for a cross --- a run plus two opposite branches.
- `bsppfit_elbow_ports()` – Port list for an elbow --- two ports at an angle.
- `bsppfit_reducer_ports()` – Port list for a reducer --- two coaxial ports of different size.

- `bsppfit_plug_ports()` – Port list for a plug's thread --- one open male port.
- `bsppfit_manifold_ports()` – Port list for a manifold --- a run plus a family of branches.
- `bsppfit_yfan_ports()` – Port list for a Y-fan --- a run plus angled branches, clearance-enforced.

6. Section: Branch-junction fillet (FDM strength reinforcement)

- `bsppfit_branch_fillet()` – A reinforced branch-to-run junction (FDM strength fillet). [Geom]

7. Section: Decorated Fittings

- `bsppfit_hexbush()` – A BSPP hex bush --- a hex-gripped reducer. [Geom]
- `bsppfit_hosetail()` – A BSPP hose tail --- a threaded port with a barbed hose end. [Geom]
- `bsppfit_union()` – A BSPP union --- two threaded ports joined by a ring nut. [Geom]
- `bsppfit_cap()` – A BSPP cap --- a female socket closed by a wrench head. [Geom]
- `bsppfit_plug()` – A BSPP plug --- a male thread closed by a wrench head. [Geom]

8. Section: Catalogue Fittings

- `bsppfit_reducer()` – A BSPP reducer --- two coaxial ports of different size. [Geom]
- `bsppfit_pipe()` – A straight BSPP pipe fitting --- two coaxial same-size ports. [Geom]
- `bsppfit_nipple()` – A BSPP nipple --- same-size ports with a wrench-grip middle. [Geom]
- `bsppfit_elbow()` – A BSPP elbow --- two same-size ports meeting at an angle. [Geom]
- `bsppfit_tee()` – A BSPP tee --- a straight run with one branch. [Geom]
- `bsppfit_cross()` – A BSPP cross --- a straight run with two opposite branches. [Geom]
- `bsppfit_manifold()` – A BSPP manifold --- a run carrying a family of branches. [Geom]

Section: Size and Dimension Tables

The BSPP dimension table and the lookups over it: nominal size, thread designator, thread major and minor diameters, pitch, useful thread length, tolerance defaults, the safe-wall and safe-middle guards, the bore heuristic, and the size and designator listings.

Function: `bsppfit_lookup()`

Synopsis: The dimension-table row for a BSPP size.

Topics: Dimensions

See Also: `bsppfit_designator()`, `bsppfit_sizes()`

Usage:

- `row = bsppfit_lookup(size);`

Description:

Returns the dimension-table row whose nominal size matches `size`, or `undef` if the size is not in the table. The match is exact. The named accessors below read individual fields from this row.

Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size (thread designator), e.g. 1/2.

Function: `bsppfit_designator()`

Synopsis: The "G" thread designator for a BSPP size.

Topics: Dimensions, Threading

See Also: [bsppfit_lookup\(\)](#), [bsppfit_designators\(\)](#)

Usage:

- `g = bsppfit_designator(size);`

Description:

Returns the G-prefixed thread designator (e.g. "G1/2") for the given nominal BSPP size. Errors if the size is not in the table.

Arguments:

By Position	What it does
size	Nominal BSPP size, e.g. 1/2.

Function: `bsppfit_major_od()`

Synopsis: The thread major (crest) diameter for a BSPP size.

Topics: Dimensions, Threading

See Also: [bsppfit_minor_od\(\)](#), [bsppfit_pitch\(\)](#)

Usage:

- `d = bsppfit_major_od(size);`

Description:

Returns the thread major (crest) diameter in millimetres for the given nominal BSPP size. Errors if the size is not in the table.

Arguments:

By Position	What it does
size	Nominal BSPP size, e.g. 1/2.

Function: `bsppfit_minor_od()`

Synopsis: The thread minor (root) diameter for a BSPP size.

Topics: Dimensions, Threading

See Also: [bsppfit_major_od\(\)](#)

Usage:

- `d = bsppfit_minor_od(size);`

Description:

Returns the thread minor (root) diameter in millimetres for the given nominal BSPP size. Errors if the size is not in the table.

Arguments:

By Position	What it does
size	Nominal BSPP size, e.g. 1/2.

Function: `bsppfit_pitch()`

Synopsis: The thread pitch for a BSPP size.

Topics: Dimensions, Threading

See Also: `bsppfit_major_od()`

Usage:

- `p = bsppfit_pitch(size);`

Description:

Returns the thread pitch in millimetres for the given nominal BSPP size. Errors if the size is not in the table.

Arguments:

By Position	What it does
size	Nominal BSPP size, e.g. 1/2.

Function: `bsppfit_useful_thread_length()`

Synopsis: The useful thread engagement length for a BSPP size.

Topics: Dimensions, Threading

See Also: `bsppfit_pitch()`

Usage:

- `l = bsppfit_useful_thread_length(size);`

Description:

Returns the useful thread engagement length in millimetres for the given nominal BSPP size --- the size-derived default thread length the fittings use. Errors if the size is not in the table.

Arguments:

By Position	What it does
size	Nominal BSPP size, e.g. 1/2.

Function: `bsppfit_safe_wall()`

Synopsis: A requested outer diameter clamped to a safe wall thickness.

Topics: Dimensions

See Also: `bsppfit_safe_middle_od()`

Usage:

- `od = bsppfit_safe_wall(requested_od, thread_od, [min_wall]);`

Description:

Returns `requested_od`, or a larger value if it would leave a wall thinner than `min_wall` over a thread of diameter `thread_od`. Use it to keep an outer diameter from cutting into the thread.

Arguments:

By Position	What it does
<code>requested_od</code>	The outer diameter asked for.
<code>thread_od</code>	The thread diameter the wall surrounds.

By Name	What it does
<code>min_wall</code>	Minimum acceptable wall thickness. Default: BSPPFIT_MIN_WALL

Function: `bsppfit_default_bore()`

Synopsis: The default through-bore diameter for a BSPP size.

Topics: Dimensions

See Also: `bsppfit_minor_od()`

Usage:

- `d = bsppfit_default_bore(size, [factor]);`

Description:

Returns the default through-bore diameter for the given nominal BSPP size, a commercial-fitting heuristic of `factor` times the thread minor diameter. Errors if the size is not in the table.

Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size, e.g. 1/2.

By Name	What it does
<code>factor</code>	Bore-to-minor-diameter ratio. Default: BSPPFIT_DEFAULT_BORE_FACTOR

Function: `bsppfit_safe_middle_od()`

Synopsis: A requested middle-band diameter clamped to a safe wall.

Topics: Dimensions

See Also: `bsppfit_safe_wall()`

Usage:

- `od = bsppfit_safe_middle_od(requested, size, [min_wall]);`

Description:

Returns **requested**, or a larger value if it would leave a wall thinner than **min_wall** over the thread of the given size. The middle-band specialisation of `bsppfit_safe_wall()`, used by the nipple grip.

Arguments:

By Position	What it does
requested	The middle-band outer diameter asked for.
size	Nominal BSPP size of the thread the band surrounds.

By Name	What it does
min_wall	Minimum acceptable wall thickness. Default: 0.5

Function: `bsppfit_sizes()`

Synopsis: The list of supported nominal BSPP sizes.

Topics: Dimensions

See Also: `bsppfit_designators()`

Usage:

- `sizes = bsppfit_sizes();`

Description:

Returns the list of supported nominal BSPP sizes, in table order --- useful for iterating over the whole range, e.g. in a gallery.

Function: `bsppfit_designators()`

Synopsis: The list of supported "G" thread designators.

Topics: Dimensions, Threading

See Also: `bsppfit_sizes()`

Usage:

- `gs = bsppfit_designators();`

Description:

Returns the list of supported G-prefixed thread designators, in table order, parallel to `bsppfit_sizes()`.

Section: Diagnostics and Utilities

The echo-only self-test, the render-variant membership test used by the per-model render dispatch, and the parametric half-cut for inspecting internal geometry.

Module: `bsppfit_cutaway()`

Synopsis: Section the child geometry open on one side, for inspection. [Geom]

Topics: Diagnostics

See Also: `bsppfit()`

Usage:

- `bsppfit_cutaway(direction) CHILDREN;`

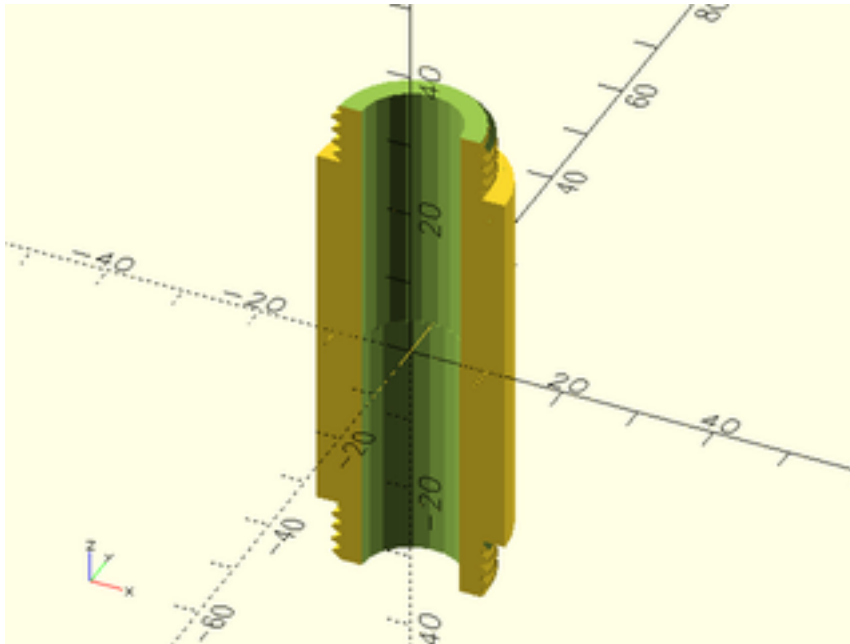
Description:

An operator that sections its child geometry, removing the half on the given side so the internals (bore, threads) show. `undef` passes the child through unchanged. Every fitting takes a `cutaway` argument that applies this for you; use the operator directly to cut arbitrary geometry.

Arguments:

By Position	What it does
<code>direction</code>	A BOSL2 direction constant (e.g. <code>FRONT</code> , <code>LEFT</code>) naming the half to remove, or <code>undef</code> for no cut.

Example 1: A pipe cut open at the front.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cutaway(FRONT) bsppfit_pipe();
```

Section: Port Representation

A fitting is described as a list of ports; a port is [direction, gender, size, termination, offset?]. These accessors and validators keep the geometry code free of raw list indexing, so the representation can change without touching the solids.

Function: `bsppfit_arm_od()`

Synopsis: The default arm outer diameter for a BSPP size.

Topics: Dimensions

See Also: `bsppfit_major_od()`

Usage:

- `d = bsppfit_arm_od(size);`

Description:

Returns the default arm outer diameter for the given nominal BSPP size: the thread major diameter plus 4 mm, kept wall-safe. This is the branch-fitting convention the super-class adopts; it is overridable per call through the `arm_od` parameter of the fittings.

Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size, e.g. 1/2.

Section: The Fitting Super-class

`bsppfit()` is the single code path behind every catalogue fitting: it takes a port list and builds the corresponding solid (arms, threads, bore, junctions). The named wrappers further down are thin facades over it.

Module: `bsppfit()`

Synopsis: The fitting super-class --- builds any BSPP fitting from a port list. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_pipe()`, `bsppfit_manifold()`

Usage:

- `bsppfit([ports], [hub_factor=], [arm_length=], [arm_od=], [bore=], [thread_length=], ...) [ATTACHMENTS];`

Description:

The single code path behind every catalogue fitting. Given a list of ports, it grows a threaded arm from a central hub along each port's direction and runs a bore from each open port through the hub, so one module makes pipes, elbows, tees, crosses, reducers and manifolds alike --- the named wrappers are thin facades that build a port list and call this. Call it directly for a topology the wrappers do not cover.

Each port record has the form `[direction, gender, size, termination]`, with an optional fifth `offset` element:

- **direction:** a unit 3-vector the arm points along, e.g. `[0,0,1]` for the +Z end.
- **gender:** "M" for an external thread, "F" for an internal socket.
- **size:** the nominal BSPP size, i.e. the thread designator, e.g. 1/2.
- **termination:** "open" for a threaded mouth with a through bore, "closed" for a thread-less rounded seal (a blanked end --- a header or dead-leg, not a cap).
- **offset:** optional 3-vector shifting the port off the hub centre, for eccentric reducers and spread manifolds.

The four per-port dimensions (`arm_length`, `arm_od`, `bore`, `thread_length`) each accept `undef` (size-derived), a single number applied to every port, or a list with one value per port. The module is attachable, anchoring on a cylindrical envelope around the arms.

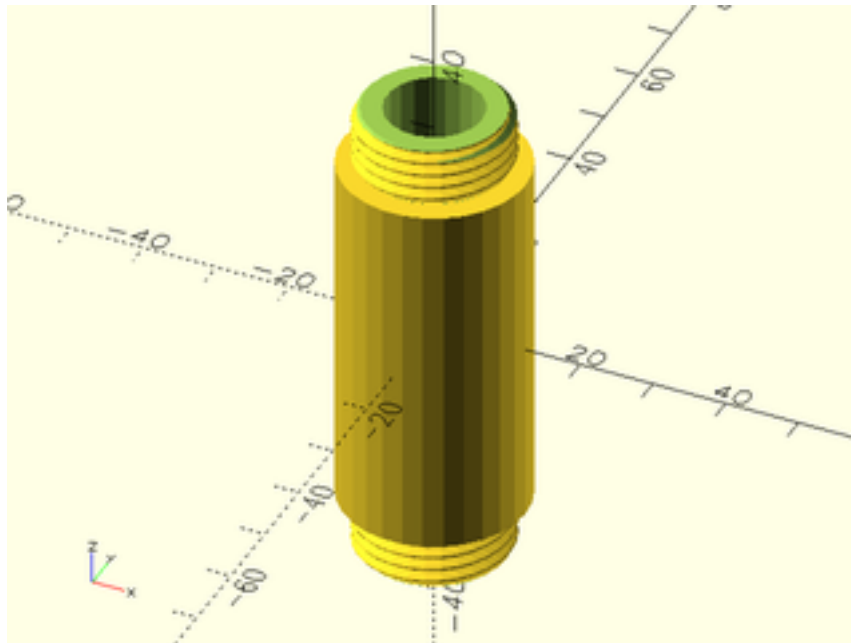
Arguments:

By Position	What it does
<code>ports</code>	List of port records, one per opening, each as described above. Default: a straight two-port pipe

By Name	What it does
<code>hub_factor</code>	Hub diameter as a multiple of the largest arm outer diameter. Default: 1.0
<code>arm_length</code>	Arm length: <code>undef</code> (size-derived thread length), a number, or one per port. Default: <code>undef</code>
<code>arm_od</code>	Arm outer diameter: <code>undef</code> (size-derived), a number, or one per port. Default: <code>undef</code>
<code>bore</code>	Through-bore diameter: <code>undef</code> (size default), a number, or one per port. Default: <code>undef</code>
<code>thread_length</code>	Thread engagement length: <code>undef</code> (size-derived), a number, or one per port. Default: <code>undef</code>
<code>manifold_safe</code>	If true, build the threads watertight for printing and boolean operations. Default: <code>true</code>
<code>tol_thread</code>	Thread-mask radial clearance, mm. Default: <code>BSPPFIT_TOL_THREAD</code>
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

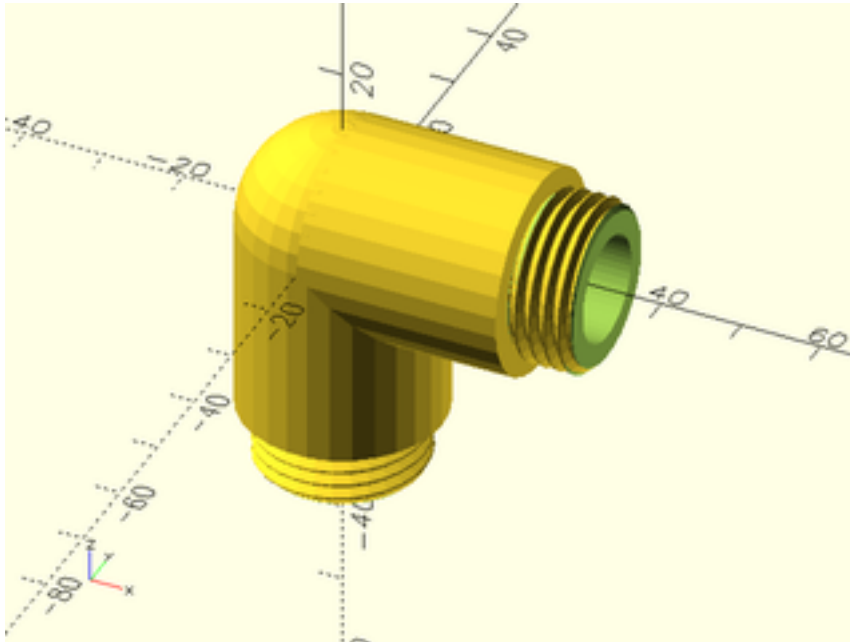
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: `BSPPFIT_UNDERSIZE` `cutaway` | `undef`, or a BOSL2 direction constant to section the solid open on that side. Default: `undef` `anchor` | Translate so the anchor point is at the origin. Default: `CENTER` `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: `UP`

Example 1: The default --- a straight male-male pipe.



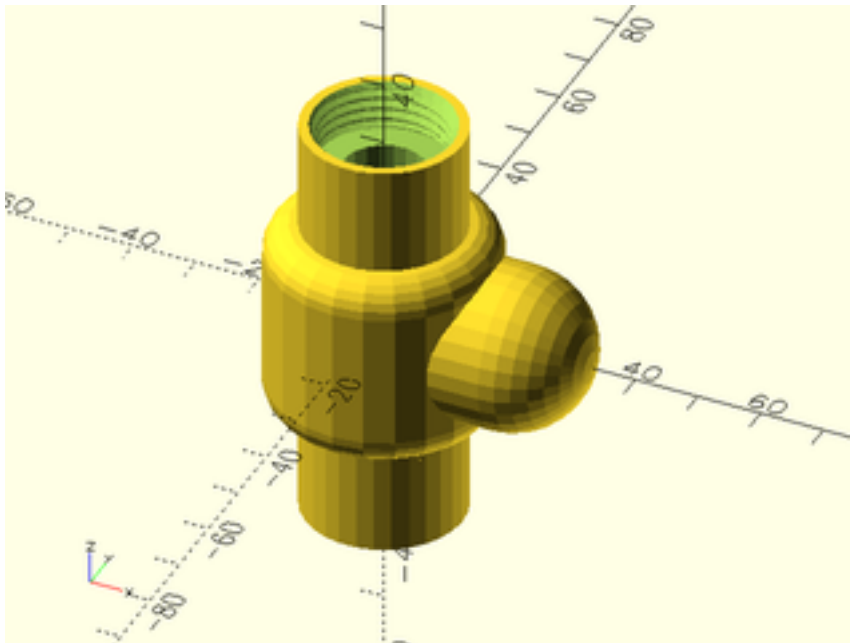
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit();
```

Example 2: A port list built by hand --- a right-angle elbow.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit([ [[0,0,-1],"M",1/2,"open"], [[1,0,0],"M",1/2,"open"] ] );
```

Example 3: A closed branch makes a dead-leg tee.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit([ [[0,0,-1],"F",1/2,"open"],
          [[0,0, 1],"F",1/2,"open"],
```

```
[[1,0, 0],"F",1/2,"closed" ]]);
```

Section: Port-List Builders

Functions that name the standard fittings as port lists, the configurations the super-class must reproduce. Each returns the port list that, fed to `bsppfit()`, produces the corresponding fitting.

Function: `bsppfit_pipe_ports()`

Synopsis: Port list for a straight pipe --- two coaxial same-size ports.

Topics: Port Lists

Usage:

- `ports = bsppfit_pipe_ports([size], [gender]);`
-

Function: `bsppfit_cap_ports()`

Synopsis: Port list for a cap's threaded socket --- one open female port.

Topics: Port Lists

Usage:

- `ports = bsppfit_cap_ports([size], [gender]);`

Description:

The mate feature of a cap: an open female port, which the super-class builds as a tapped socket with a solid back (a single port has no second mouth for the bore to reach). The sealing wrench head that makes it a finished cap is `bsppfit_cap()`'s decoration, not part of the port list. A "closed" port is NOT a cap --- it is a thread-less integral seal (a blanked end), so the cap is reproduced from an open port.

Function: `bsppfit_tee_ports()`

Synopsis: Port list for a tee --- a run plus one branch.

Topics: Port Lists

Usage:

- `ports = bsppfit_tee_ports([size], [gender]);`
-

Function: `bsppfit_cross_ports()`

Synopsis: Port list for a cross --- a run plus two opposite branches.

Topics: Port Lists

Usage:

- `ports = bsppfit_cross_ports([size], [gender]);`
-

Function: `bsppfit_elbow_ports()`

Synopsis: Port list for an elbow --- two ports at an angle.

Topics: Port Lists

Usage:

- `ports = bsppfit_elbow_ports([size], [gender], [angle]);`
-

Function: `bsppfit_reducer_ports()`

Synopsis: Port list for a reducer --- two coaxial ports of different size.

Topics: Port Lists

Usage:

- `ports = bsppfit_reducer_ports([size_a], [size_b], [gender]);`
-

Function: `bsppfit_plug_ports()`

Synopsis: Port list for a plug's thread --- one open male port.

Topics: Port Lists

Usage:

- `ports = bsppfit_plug_ports([size]);`

Description:

The mate feature of a plug: an open male port, which the super-class builds as an external thread. The solid body and the wrench head that make it a finished plug are `bsppfit_plug()`'s decoration, not part of the port list.

Function: `bsppfit_manifold_ports()`

Synopsis: Port list for a manifold --- a run plus a family of branches.

Topics: Port Lists

Usage:

- `ports = bsppfit_manifold_ports([size], [gender], [branch_count], ...);`
-

Function: `bsppfit_yfan_ports()`

Synopsis: Port list for a Y-fan --- a run plus angled branches, clearance-enforced.

Topics: Port Lists

Usage:

- `ports = bsppfit_yfan_ports([size_run], [gender_run], [size_branch], ...);`
-

Section: Branch-junction fillet (FDM strength reinforcement)

An optional strengthening fillet where a branch meets a run, OFF by default (BSPPFIT_BRANCH_FILLET). Integration step 1 (0.116.0) lands the helper and its configuration and exercises it on a test grid (render="fillet-grid" in bsppfitTest.scad); the generic body builder __bsppfit() does not yet invoke it, so every named variant is unchanged.

The external reinforcement is split by an ANGLE-CORRECTED branch/run arm ratio (an oblique branch's footprint stretches by $1/\sin(\text{lean})$, clamped at BSPPFIT_FILLET_MIN_LEAN, default 45 degrees): a reducer gets a join_prism cove flowing the branch into the run; a higher-ratio or oblique branch gets a local boss --- run-aware in radius, footprint-aware in length --- with a generous shoulder roundover (no cove --- join_prism cusps at the poles on so curved a base, and a revolved roundover floats on the saddle-shaped intersection). The policy functions (bsppfit_fillet_*) are shared so the same rule serves laterals.

Module: bsppfit_branch_fillet()

Synopsis: A reinforced branch-to-run junction (FDM strength fillet). [Geom]

Topics: Junctions, Fillets, Strength

See Also: bsppfit_tee(), bsppfit_manifold()

Usage:

- bsppfit_branch_fillet(run_size, branch_size, [angle], [enable], [run_len], [branch_len]);

Description:

A single branch cylinder joined to a run cylinder, with the optional FDM strength reinforcement applied per the chosen policy. The run lies on +Z; the branch leans **angle** from +Z in the X-Z plane (90 = a square tee). When **enable** is true the junction is reinforced. The external choice is angle-corrected (an oblique branch's footprint on the run stretches by $1/\sin(\text{lean})$, clamped at BSPPFIT_FILLET_MIN_LEAN, default 45 degrees): below the boss ratio a join_prism cove flows the branch into the run, and where the bores also differ enough an internal bore flare rounds the re-entrant edge; at or above it a local boss --- run-aware in radius and footprint-aware in length --- with a generous shoulder roundover is the reinforcement (no cove, which would cusp), the bore left sharp. When **enable** is false the plain junction is built.

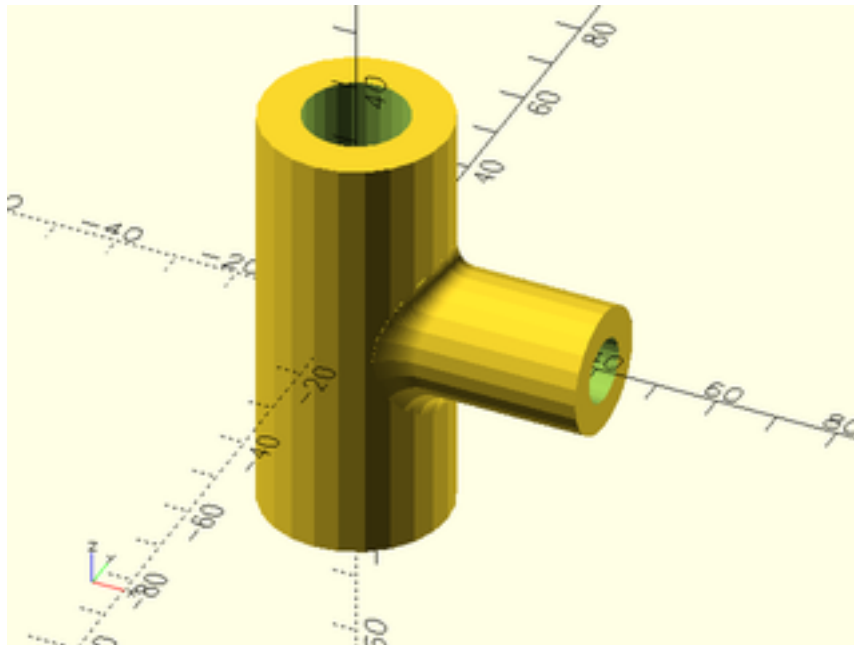
This is the integration-step-1 unit for the reinforcement: it owns its own arms and bores, so it stands alone for testing. Wiring the per-branch reinforcement into the trade fittings (tee, cross, manifold) is a later step. Exercise the grid with render="fillet-grid" in bsppfitTest.scad.

Arguments:

By Position	What it does
run_size	Nominal BSPP size of the run, e.g. 3/4.
branch_size	Nominal BSPP size of the branch, e.g. 3/8.

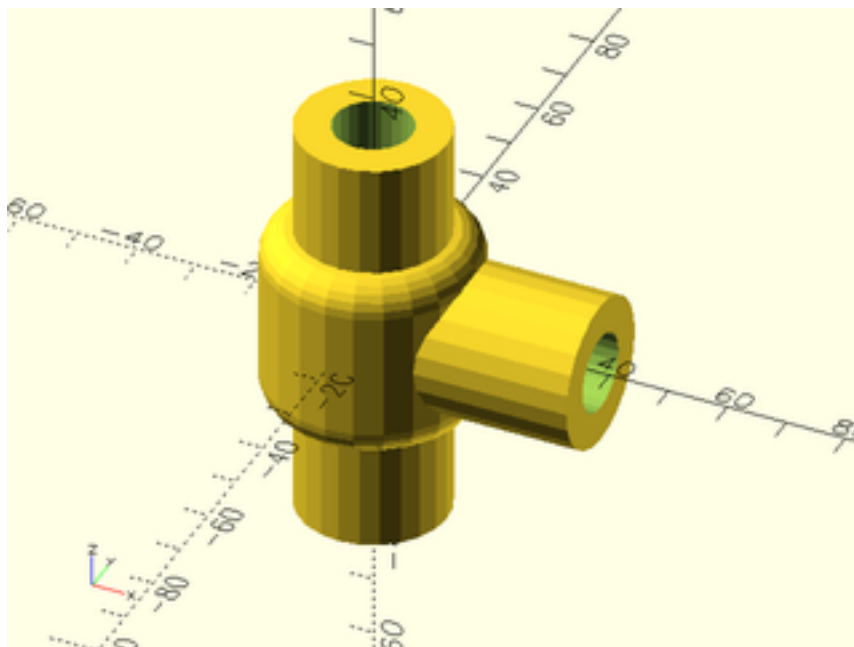
By Name	What it does
angle	Branch lean from +Z in the X-Z plane, degrees. Default: 90
enable	Apply the reinforcement. Default: BSPPFIT_BRANCH_FILLET
run_len	Run cylinder length, mm. Default: 70
branch_len	Branch cylinder length beyond the run, mm. Default: 26

Example 1: A reinforced 3/8-on-3/4 reducing junction: external cove plus internal bore flare.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_branch_fillet(3/4, 3/8, enable=true);
```

Example 2: A same-size junction: the boss carries the external fillet, the bore stays sharp.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_branch_fillet(1/2, 1/2, enable=true);
```

Section: Decorated Fittings

Fittings that add a bespoke surface to the plain super-class body: the hex-bush wrench grip, the hose-tail barbs, the union nut, and the cap and plug closures with their wrench heads.

Module: `bsppfit_hexbush()`

Synopsis: A BSPP hex bush --- a hex-gripped reducer. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_reducer()`, `bsppfit_nipple()`

Usage:

- `bsppfit_hexbush(size_male, size_female, [hex_height=], [hex_od=], ...) [ATTACHMENTS];`

Description:

A BSPP hex bush: a reducer whose body is a hexagonal prism --- the prism is the wrench grip. Two collinear opposite-gender ports of different size sit on the Z axis: the larger male thread at the +Z tip and the smaller female socket recessed in the -Z hex face, by the trade convention. The through-bore steps between the two ports' own diameters rather than running one uniform width: whichever port has the shorter thread (ties favour the female) gets a bore of its own diameter for exactly its own thread length, measured from its own face inward; the other port's diameter fills the remainder of the body, however far that reaches. This matches commercial bushings, where the narrower port's own bore does not widen out just because the body continues past it. The hex height and across-flats are size-derived unless overridden. The module is attachable.

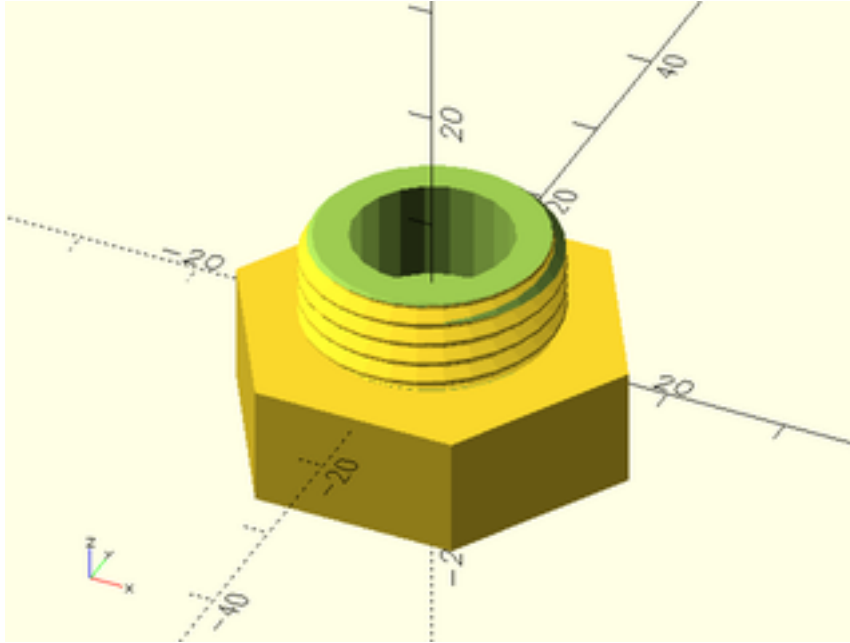
Arguments:

By Position	What it does
<code>size_male</code>	Nominal BSPP size of the male (+Z) thread. Default: 1/2
<code>size_female</code>	Nominal BSPP size of the female (-Z) socket. Default: 1/4

By Name	What it does
<code>hex_height</code>	Height of the hex body. Default: undef (size-derived)
<code>hex_od</code>	Across-flats of the hex body. Default: undef (size-derived)
<code>bore_male</code>	Through-bore diameter on the male side. Default: undef (size default)
<code>bore_female</code>	Through-bore diameter on the female side. Default: undef (size default)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: <code>BSPPFIT_TOL_THREAD</code>
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

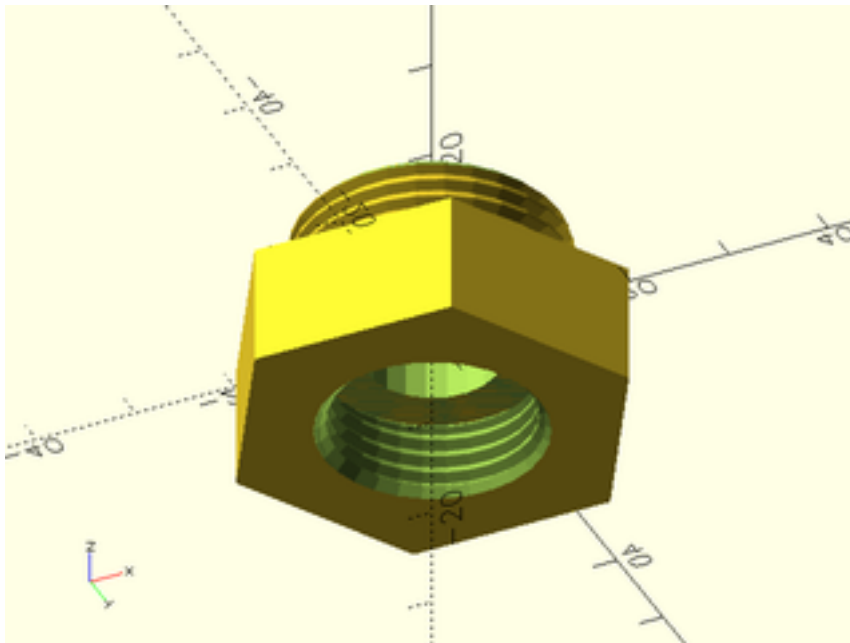
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: `BSPPFIT_UNDERSIZE` cutaway | If set to a side (e.g. `FRONT`), remove that half for inspection. Default: undef `anchor` | Translate so the anchor point is at the origin. Default: `CENTER` `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: `UP`

Example 1: A 1/2-male to 1/4-female hex bush.



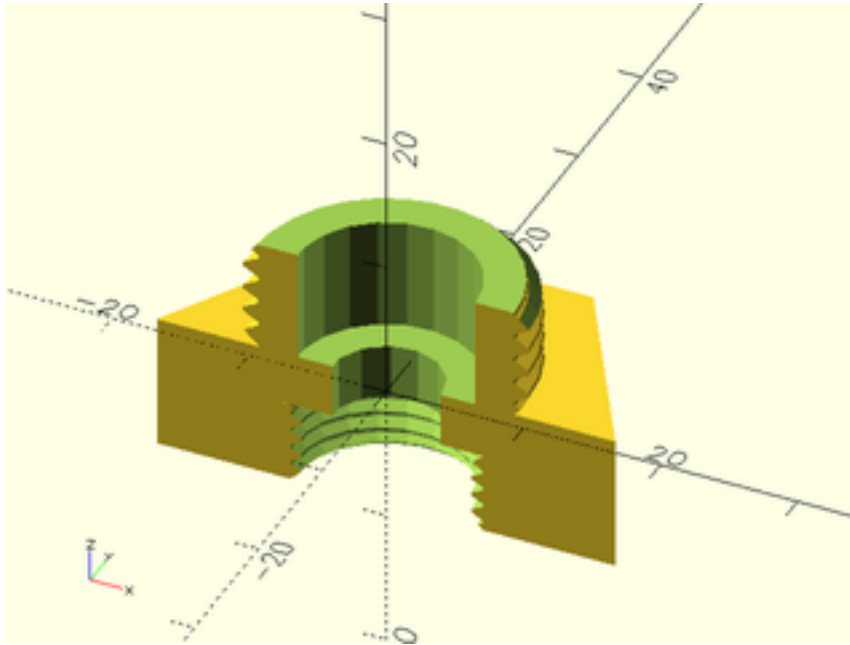
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hexbush();
```

Example 2: The same 3/4-to-1/2 bush seen from the female (1/2) end, showing the tapped socket.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hexbush(size_male=3/4, size_female=1/2);
```

Example 3: Cutaway showing the stepped bore (narrow behind the female thread, wider behind the male's).



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hexbush(cutaway=FRONT);
```

Module: `bsppfit_hosetail()`

Synopsis: A BSPP hose tail --- a threaded port with a barbed hose end. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_pipe()`

Usage:

- `bsppfit_hosetail(size, [gender=], [barb_profile=], [body_shape=], [barb_count=], ...)` [ATTACHMENTS];

Description:

A BSPP hose tail: one threaded port and one barbed-cone end that grips the inside of a flexible hose. The barb stack stands in for a thread at that end. Each barb's shape is set by `barb_profile` --- "shark_fin" (a single tapered frustum, the default), "stepped" (a frustum with a short straight land), or "christmas_tree" (barb diameters tapering toward the tip for progressive insertion). The plain body between the thread and the barbs is a round cylinder ("round", default) or a wrench-grip hex prism ("hex"), set by `body_shape`. `barb_count` barbs sit at `barb_pitch` spacing; barb diameters, body and thread lengths are size-derived unless overridden. The module is attachable.

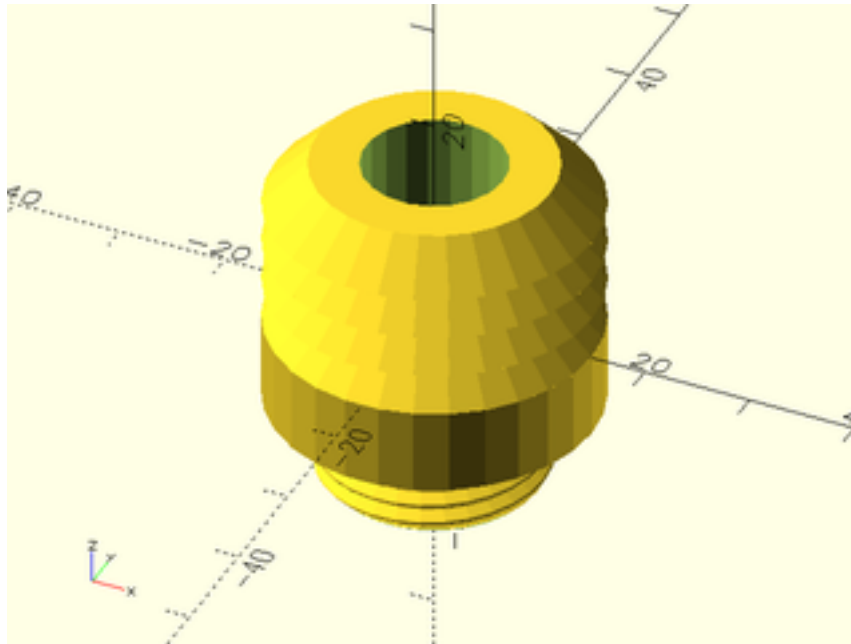
Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size of the threaded port. Default: 1/2

By Name	What it does
<code>gender</code>	Gender of the threaded port, "M" or "F". Default: "M"
<code>body_shape</code>	Plain-body cross-section, "round" or "hex". Default: "round"
<code>thread_length</code>	Length of the threaded port. Default: undef (size-derived)
<code>body_length</code>	Length of the plain body. Default: 8.0
<code>shoulder_length</code>	Length of a shoulder between body and barbs. Default: 0.0
<code>barb_count</code>	Number of barbs in the stack. Default: 4
<code>barb_pitch</code>	Spacing between barbs. Default: 4.0
<code>barb_minor_d</code>	Minor (valley) diameter of the barbs. Default: undef (size-derived)
<code>barb_major_d</code>	Major (crest) diameter of the barbs. Default: undef (size-derived)
<code>barb_profile</code>	Per-barb shape: "shark_fin", "stepped" or "christmas_tree". Default: "shark_fin"
<code>arm_od</code>	Plain-body outer diameter. Default: undef (size-derived)
<code>bore</code>	Through-bore diameter. Default: undef (size default)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

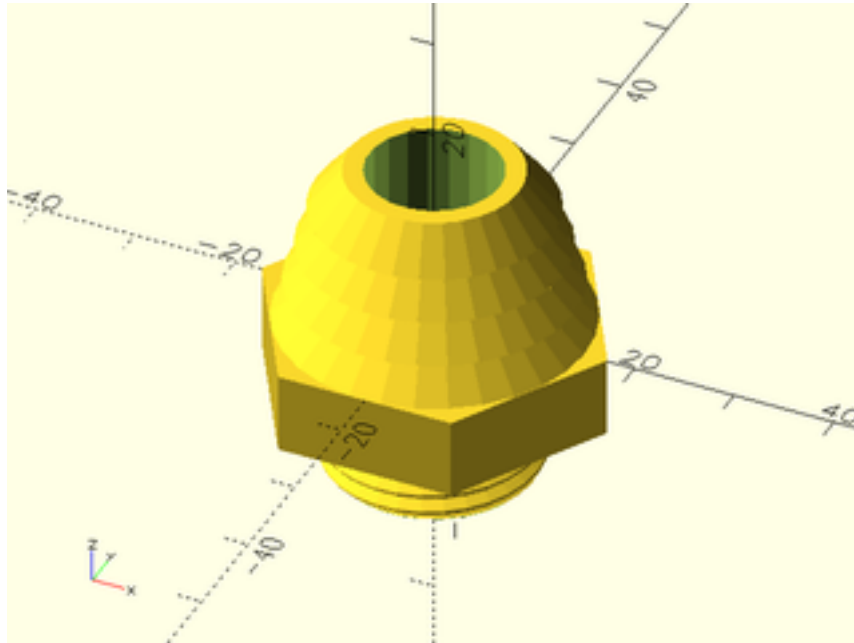
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef anchor | Translate so the anchor point is at the origin. Default: CENTER spin | Rotate this many degrees around the Z axis, after anchoring. Default: 0 orient | Vector to rotate the top toward, after spin. Default: UP

Example 1: A male 1/2 hose tail with four shark-fin barbs.



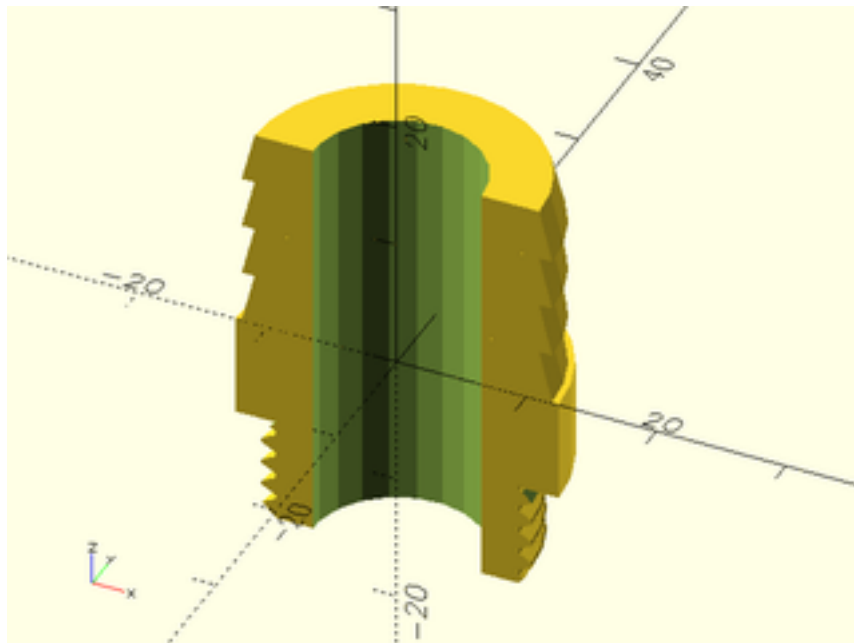
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hosetail(barb_major_d = 30);
```

Example 2: A hex-bodied hose tail with christmas-tree barbs.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hosetail(body_shape="hex", barb_profile="christmas_tree", barb_major_d = 30);
```

Example 3: Cutaway through the bore and barb stack.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_hosetail(cutaway=FRONT);
```

Module: `bsppfit_union()`

Synopsis: A BSPP union --- two threaded ports joined by a ring nut. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_pipe()`, `bsppfit_nipple()`

Usage:

- `bsppfit_union(size, [gender_a=], [gender_b=], [explode=], ...) [ATTACHMENTS];`

Description:

A BSPP union: two coaxial threaded ports joined by a ring nut, so the halves part without rotating the pipes. Built as the real three-part assembly --- two identical threaded tails and a hex ring nut, threaded over its full length, that draws them together against a 60-degree cone seat. (An earlier version gave tail B a plain shoulder retained by a narrower nut lip, matching the dedicated model; that shape cannot be assembled from three separately printed parts --- the lip has to be narrower than the shoulder to retain it, which makes it too narrow for the shoulder to pass through during assembly. Both tails now carry the same coupling-thread collar as tail A always did, and the nut is threaded throughout; see the Union History note.) Each port's gender is set independently. With **explode** greater than zero the two tails pull apart from the nut for an assembly view. The coupling thread, body and collar lengths are adjustable. The module is attachable.

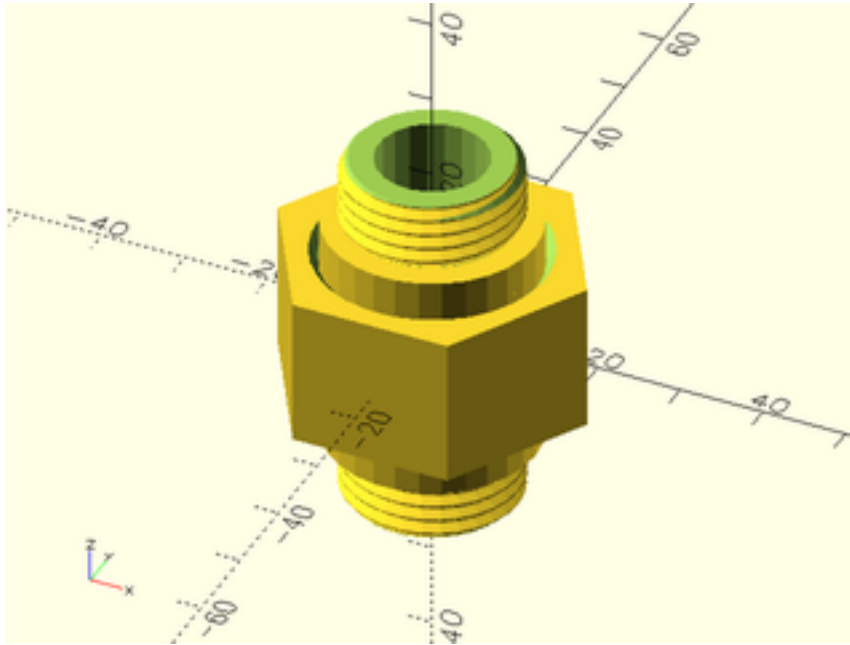
Arguments:

By Position	What it does
size	Nominal BSPP size (both ports). Default: 1/2

By Name	What it does
gender_a	Gender of the -Z port, "M" or "F". Default: "M"
gender_b	Gender of the +Z port, "M" or "F". Default: "M"
coupling_size	BSPP size of the nut's coupling thread. Default: undef (size-derived)
thread_length	Length of each pipe-end thread/socket. Default: undef (size-derived)
body_length	Plain body length of each tail. Default: 6.0
collar_length	Collar (coupling-thread) length of each tail. Default: 8.0
arm_od	Plain body outer diameter. Default: undef (major_od + 4)
bore	Through-bore diameter. Default: undef (size default)
explode	Separation of the two tails from the nut, mm; 0 is assembled. Default: 0
manifold_safe	If true, nudge coincident faces apart so the result stays manifold. Default: true
tol_thread	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
undersize	Amount to decrease the male thread diameters, in mm, without changing

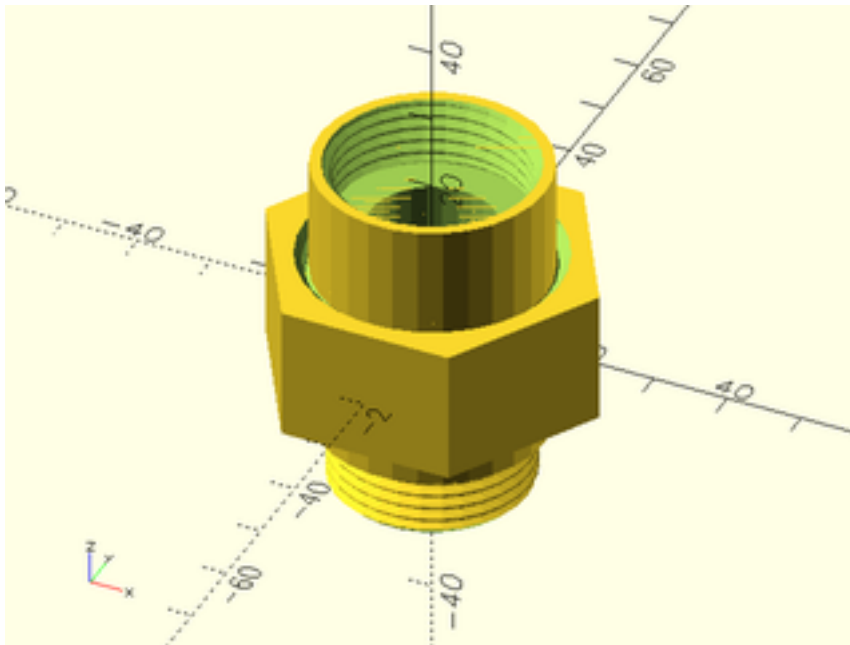
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see **tol_thread**). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef **anchor** | Translate so the anchor point is at the origin. Default: CENTER **spin** | Rotate this many degrees around the Z axis, after anchoring. Default: 0 **orient** | Vector to rotate the top toward, after spin. Default: UP

Example 1: A male-male 1/2 union.



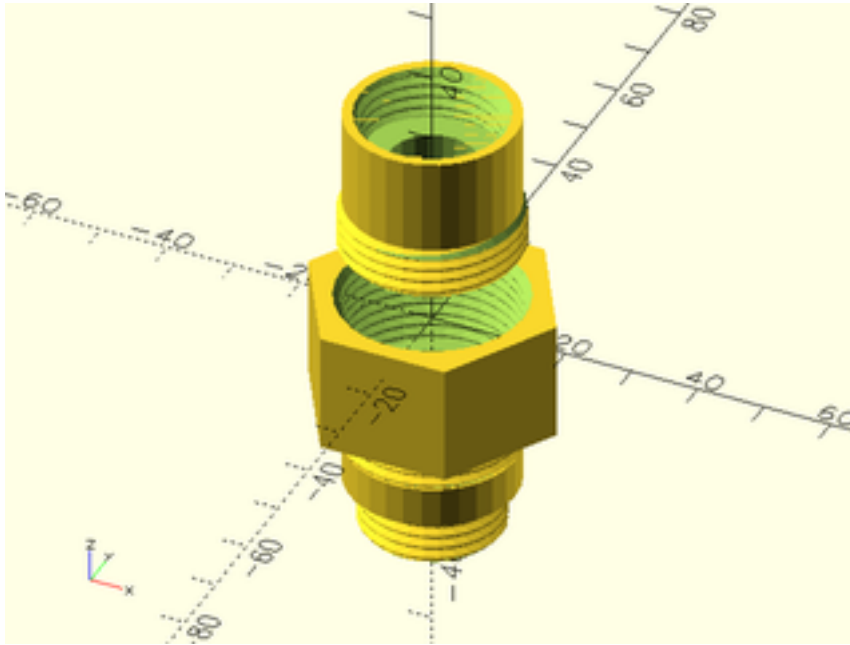
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_union();
```

Example 2: A male-to-female union in 3/4.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_union(size=3/4, gender_a="M", gender_b="F");
```

Example 3: An exploded view of the three parts.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_union(gender_a="M", gender_b="F", explode=12);
```

Module: `bsppfit_cap()`

Synopsis: A BSPP cap --- a female socket closed by a wrench head. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_plug()`

Usage:

- `bsppfit_cap(size, [head_height=], [head_shape=], ...)` [ATTACHMENTS];

Description:

A BSPP cap: a female socket closed at the back by a solid head that carries a wrench grip. `head_shape` "hex" (the commercial norm, the default) gives a hexagonal head; "round" a plain domed boss. The head height and across-flats or diameter are size-derived unless overridden. The module is attachable.

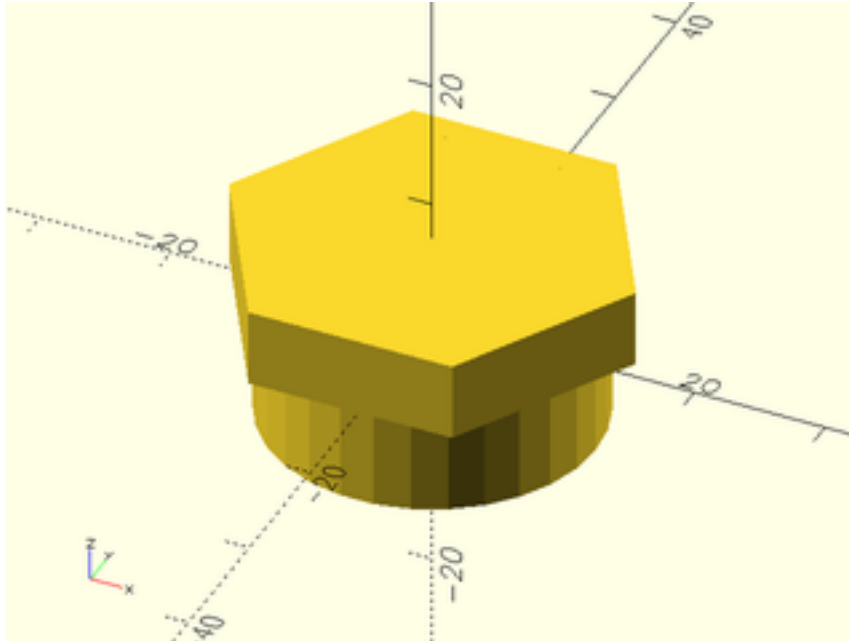
Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size of the socket. Default: 1/2

By Name	What it does
<code>head_height</code>	Height of the closing head. Default: 6
<code>head_od</code>	Across-flats or diameter of the head. Default: undef (size-derived)
<code>head_shape</code>	Head shape, "hex" or "round". Default: "hex"
<code>bore</code>	Internal (blind) bore diameter. Default: undef (size default)

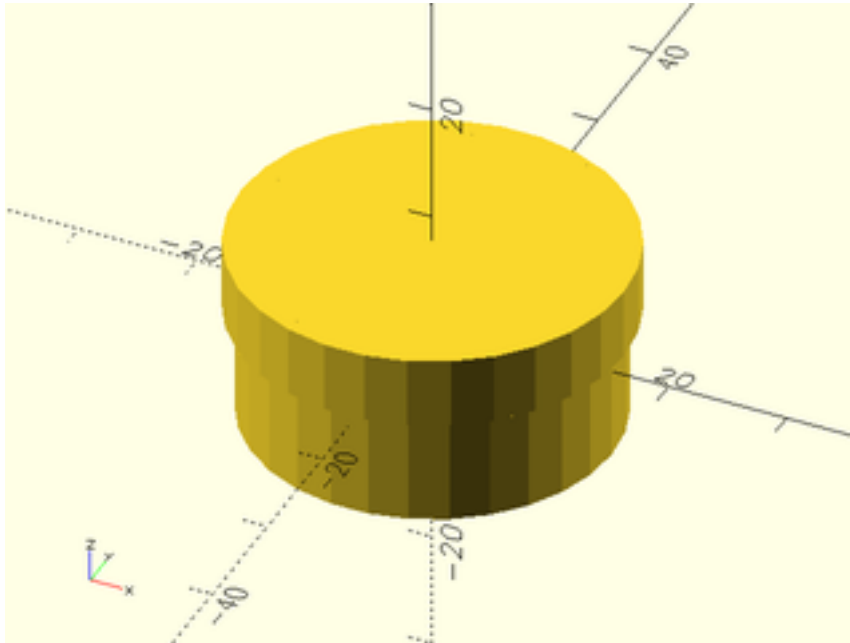
By Name	What it does
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>cutaway</code>	If set to a side (e.g. FRONT), remove that half for inspection. Default: undef
<code>anchor</code>	Translate so the anchor point is at the origin. Default: CENTER
<code>spin</code>	Rotate this many degrees around the Z axis, after anchoring. Default: 0
<code>orient</code>	Vector to rotate the top toward, after spin. Default: UP

Example 1: A 1/2 hex-headed cap.



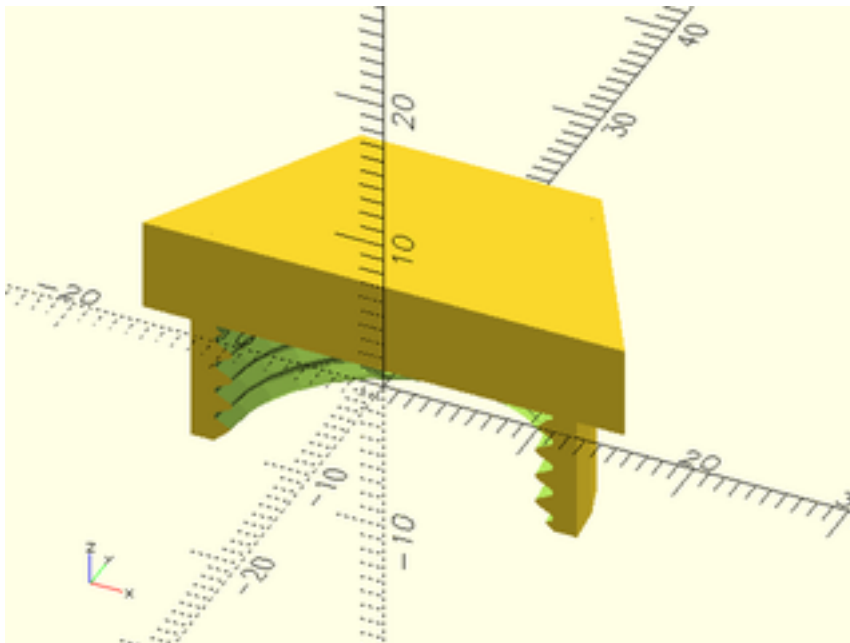
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cap();
```

Example 2: A round-headed cap in 3/4.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cap(size=3/4, head_shape="round");
```

Example 3: Cutaway showing the socket and closing head.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cap(cutaway=FRONT);
```

Module: `bsppfit_plug()`**Synopsis:** A BSPP plug --- a male thread closed by a wrench head. [Geom]**Topics:** Parts, Threading, Pipe Fittings**See Also:** `bsppfit_cap()`**Usage:**

- `bsppfit_plug(size, [head_height=], [head_shape=], ...)` [ATTACHMENTS];

Description:

A BSPP plug: a male thread closed at the back by a solid head that carries a wrench grip. `head_shape` "hex" (the default) gives a hexagonal head; "round" a plain domed boss. The head height and across-flats or diameter are size-derived unless overridden. The module is attachable.

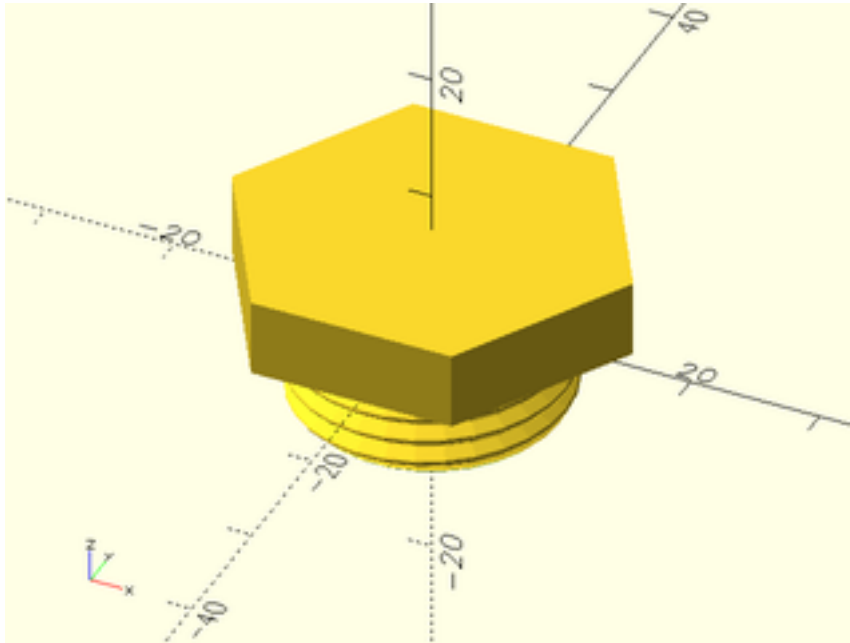
Arguments:

By Position	What it does
<code>size</code>	Nominal BSPP size of the thread. Default: 1/2

By Name	What it does
<code>head_height</code>	Height of the closing head. Default: 6
<code>head_od</code>	Across-flats or diameter of the head. Default: undef (size-derived)
<code>head_shape</code>	Head shape, "hex" or "round". Default: "hex"
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

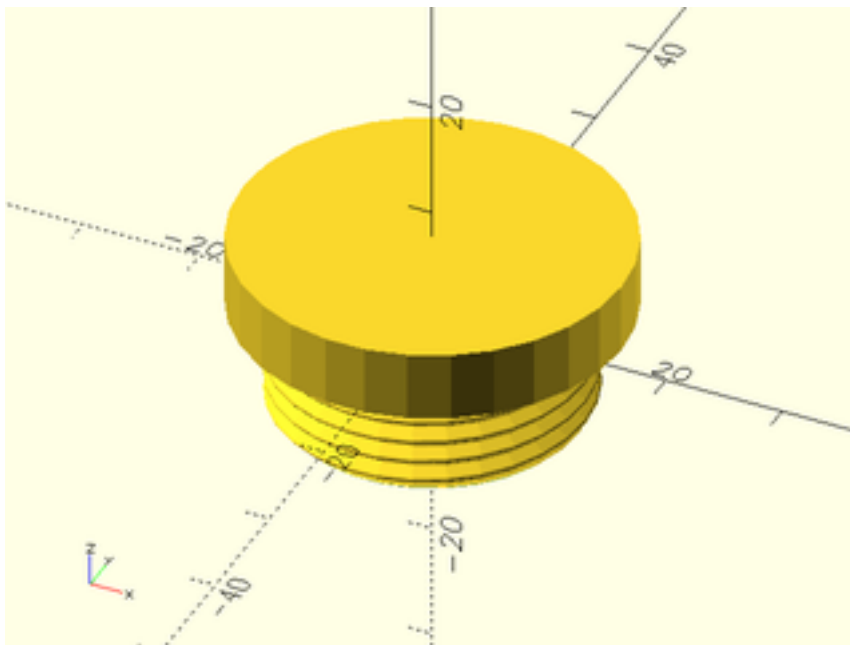
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef `anchor` | Translate so the anchor point is at the origin. Default: CENTER `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: UP

Example 1: A 1/2 hex-headed plug.



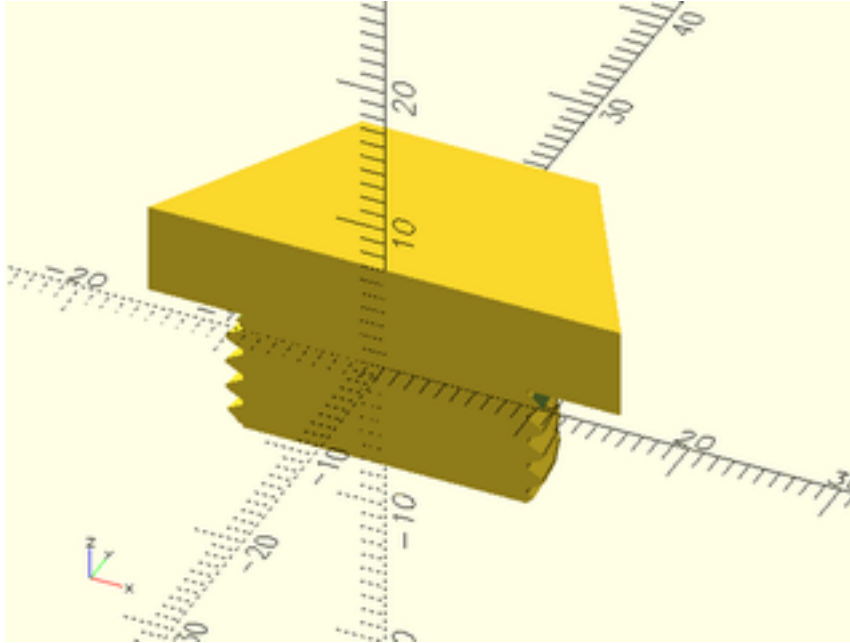
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_plug();
```

Example 2: A round-headed plug in 3/4.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_plug(size=3/4, head_shape="round");
```

Example 3: Cutaway showing the thread and closing head.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_plug(cutaway=FRONT);
```

Section: Catalogue Fittings

The named catalogue wrappers --- pipe, nipple, elbow, tee, cross, reducer and manifold --- each a thin facade over bsppfit() with a fixed port topology and convenient per-end overrides.

Module: bsppfit_reducer()

Synopsis: A BSPP reducer --- two coaxial ports of different size. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: bsppfit_pipe(), bsppfit_nipple(), bsppfit_hexbush()

Usage:

- bsppfit_reducer(size_large, size_small, [gender_large=], [gender_small=], ...) [ATTACHMENTS];

Description:

A BSPP reducer joining two coaxial ports of different nominal size, the larger on the -Z end and the smaller on the +Z end. The taper between them is formed by the super-class hub, so there is no separate transition-length parameter. Set **eccentric** to LEFT, RIGHT, FRONT or BACK to shift the small port by half the arm-OD difference, bringing the two outer walls onto a common tangent (an eccentric reducer); CENTER, the default, is concentric. **hex_ports** adds a wrench-grip hexagonal band on the "large", "small" or "both" body sections (or "none"), each band's across-flats matching that section's arm OD. The module is attachable.

Arguments:

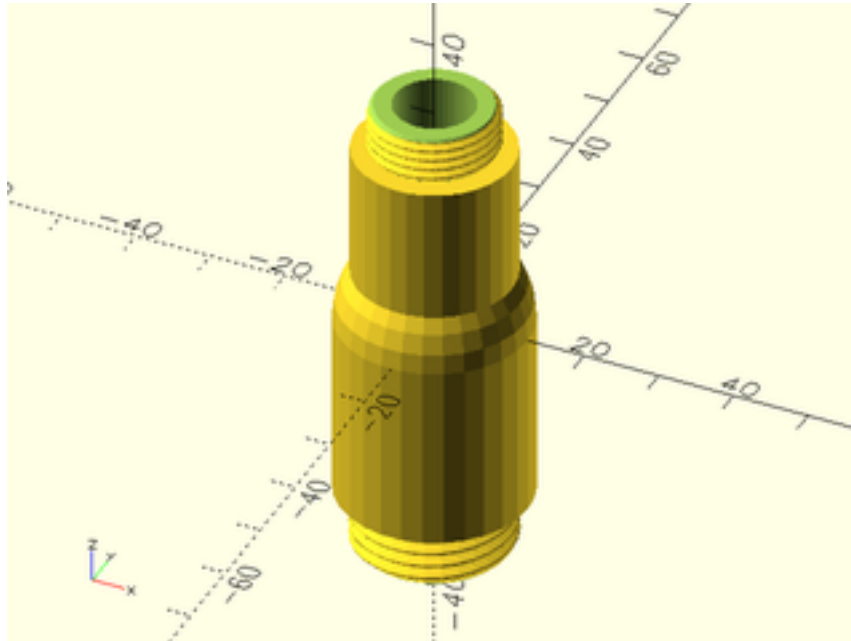
By Position	What it does
size_large	Nominal BSPP size of the -Z (large) end. Default: 1/2

By Position	What it does
<code>size_small</code>	Nominal BSPP size of the +Z (small) end. Default: 3/8

By Name	What it does
<code>gender_large</code>	Gender of the large end, "M" or "F". Default: "M"
<code>gender_small</code>	Gender of the small end, "M" or "F". Default: "M"
<code>hex_ports</code>	Which sections get a wrench-grip hex band: "none", "large", "small" or "both". Default: "none"
<code>arm_length_large</code>	Length of the large arm. Default: undef (size-derived)
<code>arm_length_small</code>	Length of the small arm. Default: undef (size-derived)
<code>arm_od_large</code>	Outer diameter of the large arm. Default: undef (size-derived)
<code>arm_od_small</code>	Outer diameter of the small arm. Default: undef (size-derived)
<code>bore_large</code>	Through-bore diameter at the large end. Default: undef (size default)
<code>bore_small</code>	Through-bore diameter at the small end. Default: undef (size default)
<code>thread_length_large</code>	Thread length at the large end. Default: undef (derived)
<code>thread_length_small</code>	Thread length at the small end. Default: undef (derived)
<code>eccentric</code>	Shift the small port onto a common wall tangent: CENTER, LEFT, RIGHT, FRONT or BACK.
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef anchor | Translate so the anchor point is at the origin. Default: CENTER spin | Rotate this many degrees around the Z axis, after anchoring. Default: 0 orient | Vector to rotate the top toward, after spin. Default: UP

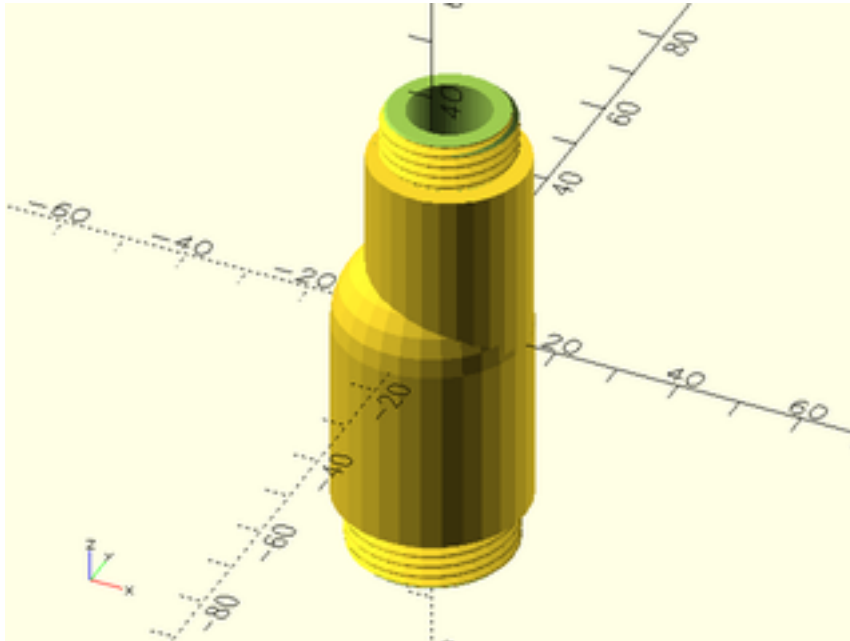
Example 1: A male-male 1/2-to-3/8 reducer.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
```

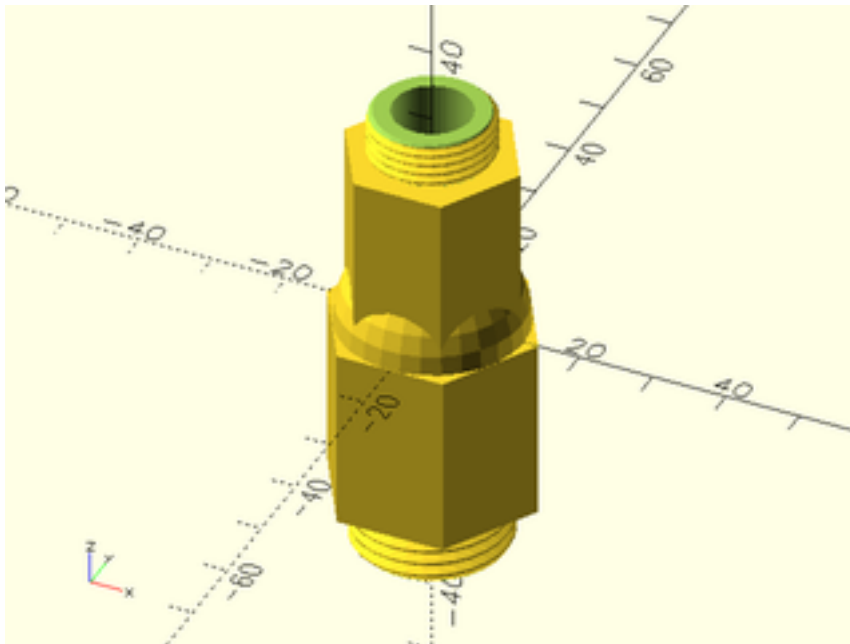
```
bsppfit_reducer();
```

Example 2: An eccentric reducer with the walls tangent on the right.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_reducer(size_large=3/4, size_small=1/2, eccentric=RIGHT);
```

Example 3: A hex-gripped reducer on both sections.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
```

```
bsppfit_reducer(hex_ports="both");
```

Module: `bsppfit_pipe()`

Synopsis: A straight BSPP pipe fitting --- two coaxial same-size ports. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_nipple()`, `bsppfit_reducer()`, `bsppfit_elbow()`

Usage:

- `bsppfit_pipe(size, [gender_a=], [gender_b=], ...) [ATTACHMENTS];`

Description:

A straight BSPP (British Standard Pipe Parallel) pipe fitting: two coaxial ports of the same nominal size, port A on the -Z end and port B on the +Z end. Each end's gender is chosen independently, so the fitting may be male-male, female-female, or a male-to-female adaptor. The end-to-end length is the sum of the two arm lengths; leave the arm lengths undefined for the size-derived default. The per-end arm length, arm outer diameter, through-bore diameter and thread length may each be overridden on their own. The module is attachable, with anchoring on a cylindrical envelope around the two arms.

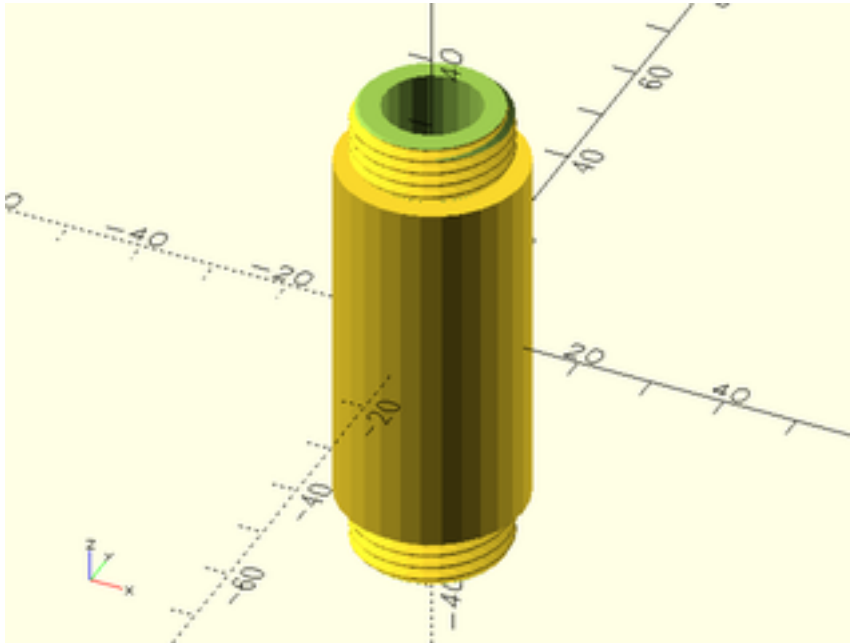
Arguments:

By Position	What it does
size	Nominal BSPP size, the thread designator (e.g. 1/2, 3/4). Default: 1/2

By Name	What it does
gender_a	Gender of the -Z end, "M" or "F". Default: "M"
gender_b	Gender of the +Z end, "M" or "F". Default: "M"
arm_length_a	Length of the -Z arm. Default: undef (size-derived)
arm_length_b	Length of the +Z arm. Default: undef (size-derived)
arm_od_a	Outer diameter of the -Z arm. Default: undef (size-derived)
arm_od_b	Outer diameter of the +Z arm. Default: undef (size-derived)
bore_a	Through-bore diameter at the -Z end. Default: undef (size default)
bore_b	Through-bore diameter at the +Z end. Default: undef (size default)
thread_length_a	Thread length at the -Z end. Default: undef (derived)
thread_length_b	Thread length at the +Z end. Default: undef (derived)
manifold_safe	If true, nudge coincident faces apart so the result stays manifold. Default: true
tol_thread	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
undersize	Amount to decrease the male thread diameters, in mm, without changing

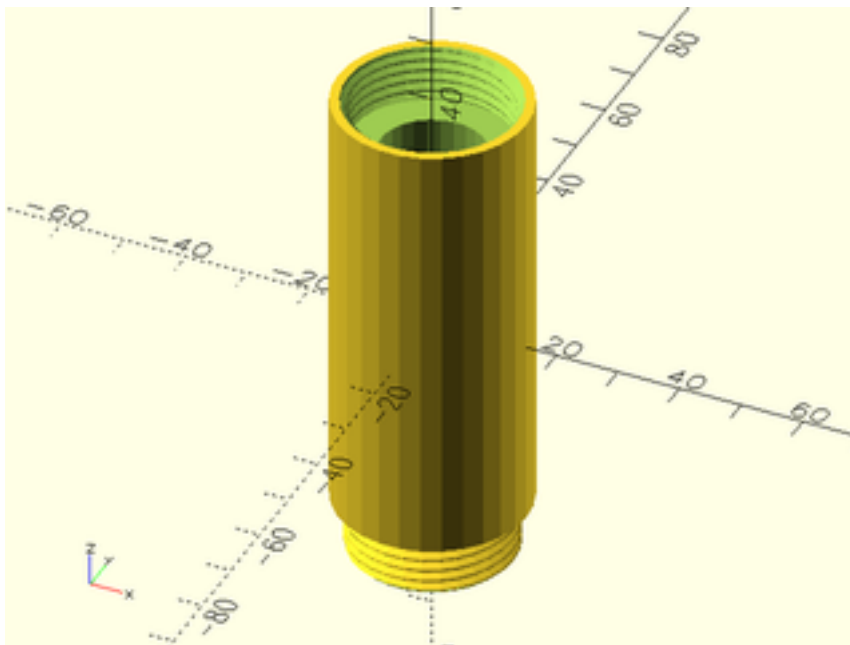
pitch or **thread depth**; negative values enlarge them. Female threads are **unaffected** (see **tol_thread**). Default: BSPPFIT_UNDERSIZE **cutaway** | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef **anchor** | Translate so the anchor point is at the origin. Default: CENTER **spin** | Rotate this many degrees around the Z axis, after anchoring. Default: 0 **orient** | Vector to rotate the top toward, after spin. Default: UP

Example 1: Male-male pipe at the default 1/2 size.



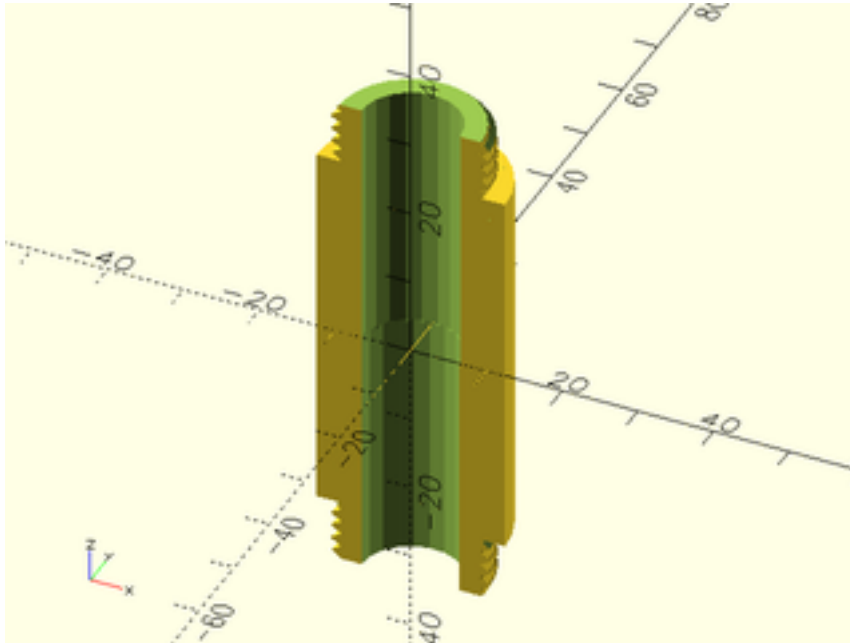
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_pipe();
```

Example 2: A male-to-female adaptor in 3/4.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_pipe(size=3/4, gender_a="M", gender_b="F");
```

Example 3: Cutaway showing the through-bore and both threads.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_pipe(cutaway=FRONT);
```

Module: `bsppfit_nipple()`

Synopsis: A BSPP nipple --- same-size ports with a wrench-grip middle. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_pipe()`, `bsppfit_reducer()`

Usage:

- `bsppfit_nipple(size, [gender_a=], [gender_b=], [middle_length=], ...) [ATTACHMENTS];`

Description:

A BSPP nipple: two coaxial same-size ports joined by a short unthreaded wrench-grip middle section --- the feature that tells a nipple from a plain pipe. The middle is a hexagonal prism (the commercial norm, `middle_shape` "hex") or a cylinder ("round"), of length `middle_length` and across-flats or diameter `middle_od` (default the thread major diameter plus 4 mm, floored to a safe wall). `middle_length` must be positive; for a plain short connector use `bsppfit_pipe()` instead. Each end's gender is independent: a male end sits thread-flush, a female end keeps a size-derived wall for its internal thread. The module is attachable.

Arguments:

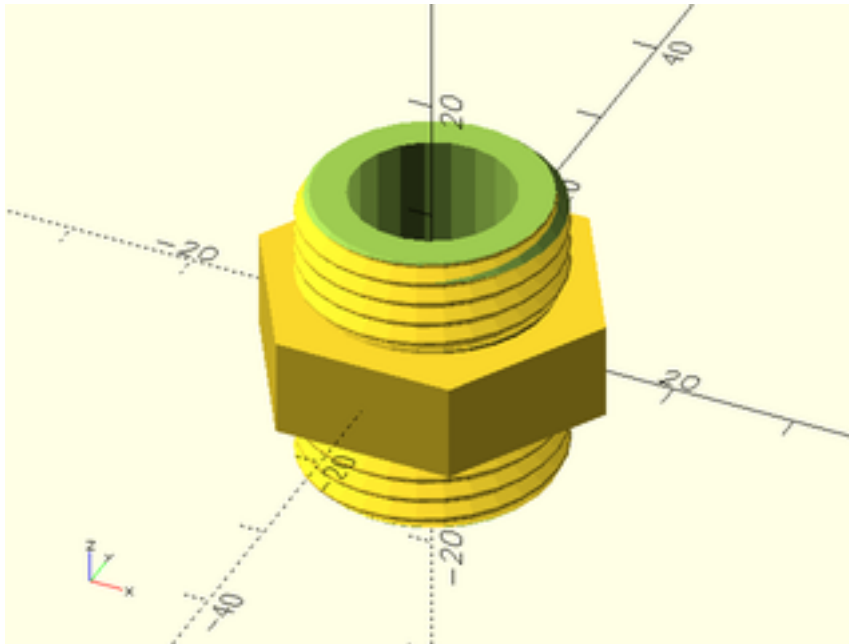
By Position	What it does
<code>size</code>	Nominal BSPP size (both ends). Default: 1/2

By Name	What it does
<code>gender_a</code>	Gender of the -Z end, "M" or "F". Default: "M"

By Name	What it does
<code>gender_b</code>	Gender of the +Z end, "M" or "F". Default: "M"
<code>thread_length_a</code>	Thread length at the -Z end. Default: undef (derived)
<code>thread_length_b</code>	Thread length at the +Z end. Default: undef (derived)
<code>middle_length</code>	Length of the unthreaded grip (must be > 0). Default: 8
<code>middle_od</code>	Grip across-flats or diameter. Default: undef (major diameter + 4 mm)
<code>middle_shape</code>	Grip shape, "hex" or "round". Default: "hex"
<code>bore_a</code>	Through-bore diameter at the -Z end. Default: undef (size default)
<code>bore_b</code>	Through-bore diameter at the +Z end. Default: undef (size default)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

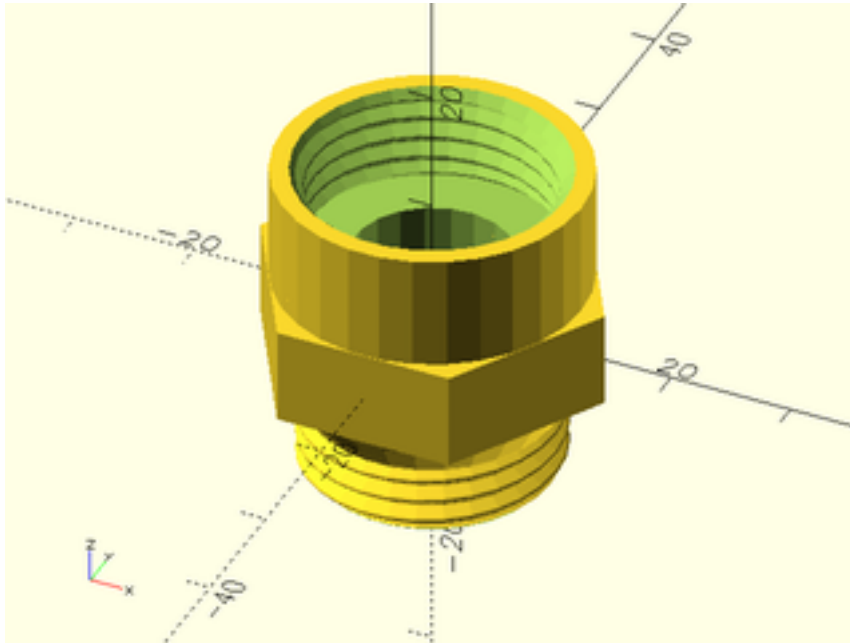
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef `anchor` | Translate so the anchor point is at the origin. Default: CENTER `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: UP

Example 1: A male-male 1/2 nipple with a hex grip.



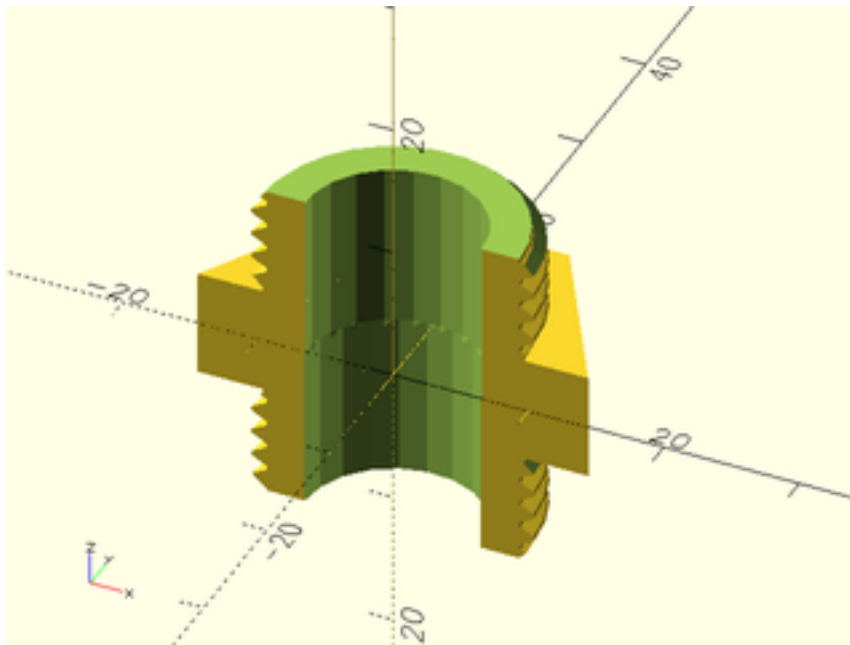
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_nipple();
```

Example 2: A male-to-female nipple (male A, female B).



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_nipple(gender_a="M", gender_b="F");
```

Example 3: Cutaway through the bore and grip.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_nipple(cutaway=FRONT);
```

Module: bsppfit_elbow()

Synopsis: A BSPP elbow --- two same-size ports meeting at an angle. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: [bsppfit_pipe\(\)](#), [bsppfit_tee\(\)](#)

Usage:

- `bsppfit_elbow(size, [angle=], [gender_a=], [gender_b=], ...)` [ATTACHMENTS];

Description:

A BSPP elbow: two same-size ports meeting at **angle** (default 90). Arm A runs along +Z; arm B leans by **angle** in the X-Z plane. The two arms are joined by a smooth swept bend --- a quarter-torus at 90 degrees, a torus arc of (180 - angle) in general --- of major radius **bend_radius**, with the bore following the bend as a swept toroidal arc of constant width. This matches commercial elbows and replaces the earlier mitred junction. Gender, arm length, arm OD, bore, thread length and bend radius are settable. The module is attachable.

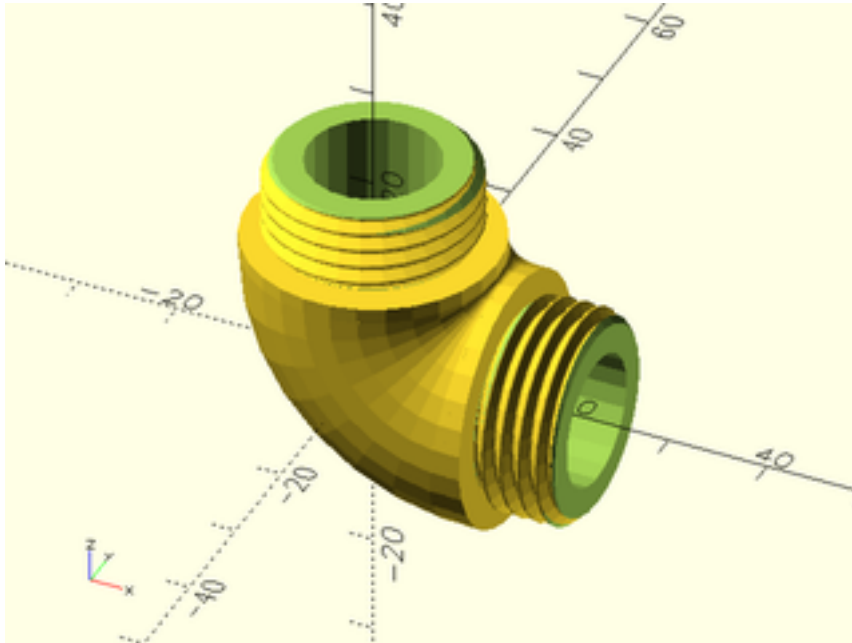
Arguments:

By Position	What it does
size	Nominal BSPP size (both ends). Default: 1/2

By Name	What it does
gender_a	Gender of the +Z arm, "M" or "F". Default: "M"
gender_b	Gender of the angled arm, "M" or "F". Default: "M"
angle	Angle between the arms, in degrees, 0 < angle < 180. Default: 90
thread_length	Length of each external thread / internal socket. Default: undef (size-derived)
arm_length	Length of each arm from the corner to its tip. Default: undef (thread_length + tangent offset + 1.5)
arm_od	Outer diameter of the arms. Default: undef (major_od + 4)
bore	Through-bore diameter. Default: undef (size default)
bend_radius	Major radius of the swept bend. Default: undef (arm_od/2 + 0.5)
manifold_safe	If true, nudge coincident faces apart so the result stays manifold. Default: true
tol_thread	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
undersize	Amount to decrease the male thread diameters, in mm, without changing

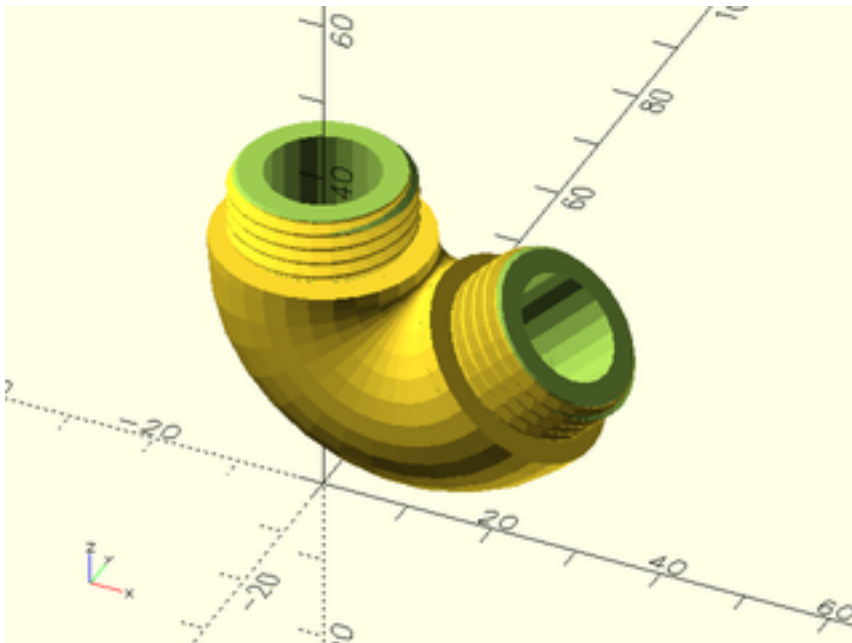
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see **tol_thread**). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef **anchor** | Translate so the anchor point is at the origin. Default: CENTER **spin** | Rotate this many degrees around the Z axis, after anchoring. Default: 0 **orient** | Vector to rotate the top toward, after spin. Default: UP

Example 1: A square (90) male-male elbow.



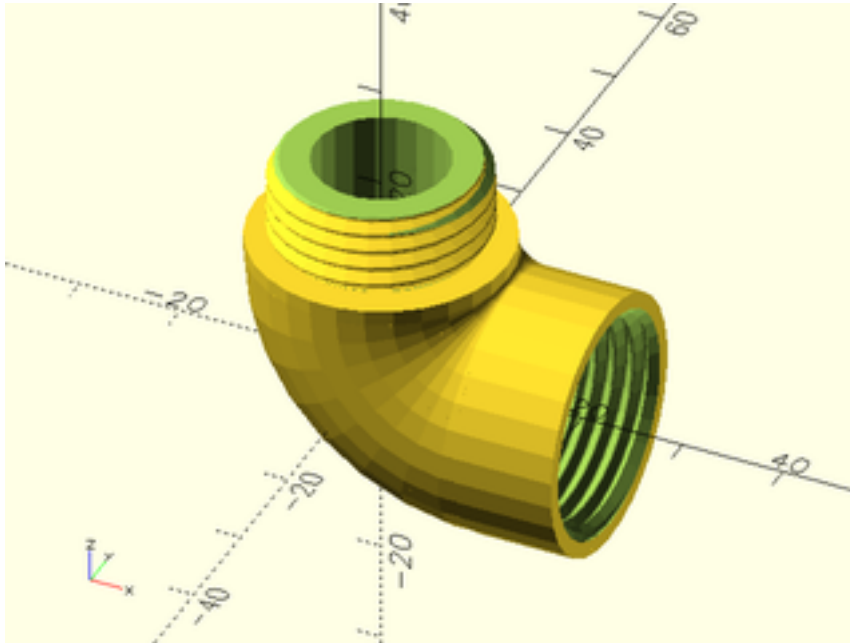
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_elbow();
```

Example 2: A 45-degree male-male elbow.



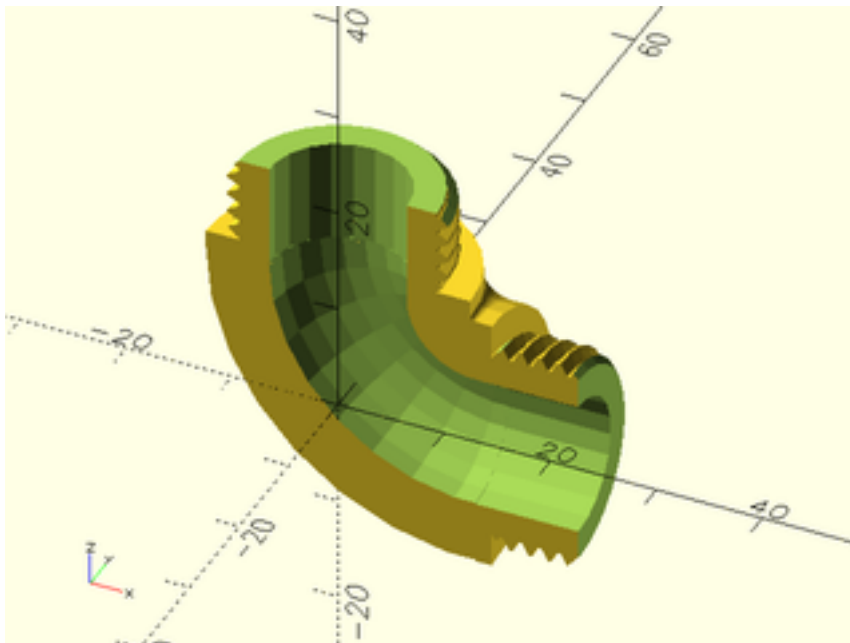
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_elbow(angle=45);
```

Example 3: A male-to-female elbow (male A, female B).



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_elbow(gender_a="M", gender_b="F");
```

Example 4: Cutaway showing the swept bore through the bend.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_elbow(cutaway=FRONT);
```

Module: `bsppfit_tee()`

Synopsis: A BSPP tee --- a straight run with one branch. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_cross()`, `bsppfit_manifold()`, `bsppfit_elbow()`

Usage:

- `bsppfit_tee(size_run, [size_branch=], [gender_run=], [branch_angle=], ...)` [ATTACHMENTS];

Description:

A BSPP tee: a straight run (two collinear ports on -Z and +Z) with one branch leaning by `branch_angle` in the X-Z plane (default 90, a square tee). The run and branch may differ in size (a reducing tee) and gender. The three-way junction --- joined arms and rounded bore inflections --- is the super-class's, so the wrapper delegates straight to it. The two run ports share their overrides; for a fully asymmetric run, call `bsppfit()` directly. The module is attachable.

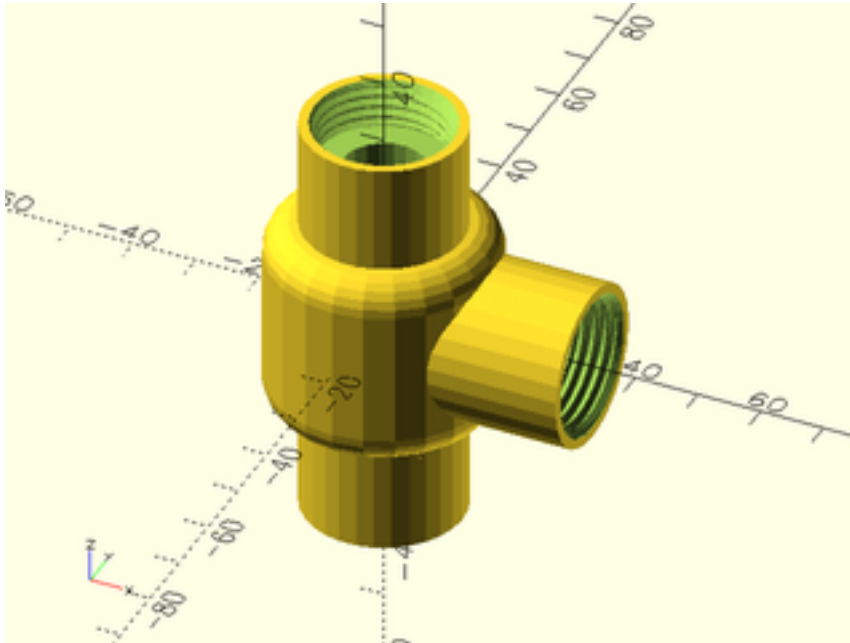
Arguments:

By Position	What it does
<code>size_run</code>	Nominal BSPP size of the run. Default: 1/2
<code>size_branch</code>	Nominal BSPP size of the branch. Default: 1/2

By Name	What it does
<code>gender_run</code>	Gender of both run ports, "M" or "F". Default: "F"
<code>gender_branch</code>	Gender of the branch port, "M" or "F". Default: "F"
<code>branch_angle</code>	Branch lean from +Z in degrees, 0..180. Default: 90
<code>arm_length_run</code>	Length of each run arm. Default: undef (size-derived)
<code>arm_length_branch</code>	Length of the branch arm. Default: undef (size-derived)
<code>arm_od_run</code>	Outer diameter of the run arms. Default: undef (size-derived)
<code>arm_od_branch</code>	Outer diameter of the branch arm. Default: undef (size-derived)
<code>bore_run</code>	Through-bore diameter of the run. Default: undef (size default)
<code>bore_branch</code>	Through-bore diameter of the branch. Default: undef (size default)
<code>thread_length_run</code>	Thread length of the run ports. Default: undef (derived)
<code>thread_length_branch</code>	Thread length of the branch. Default: undef (derived)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

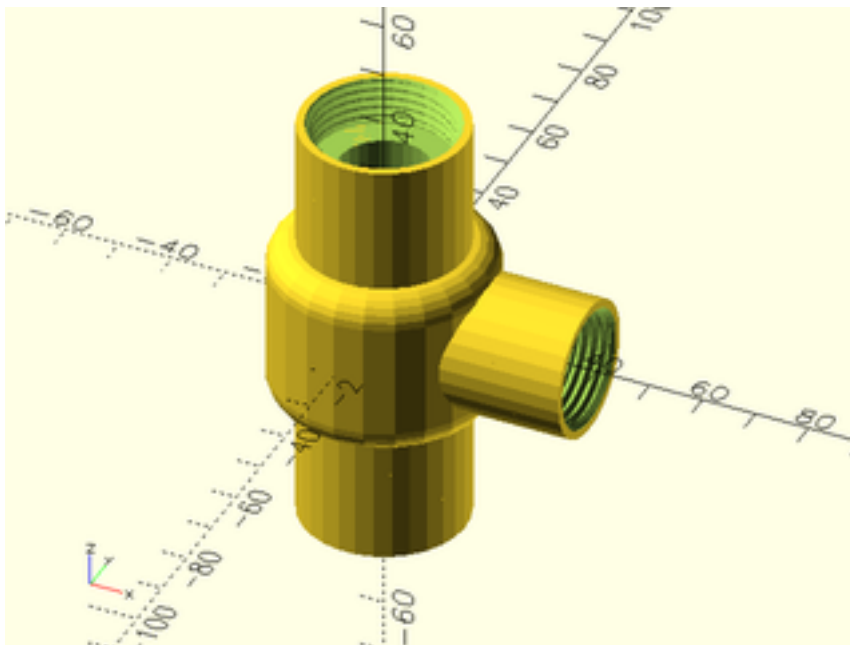
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef `anchor` | Translate so the anchor point is at the origin. Default: CENTER `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: UP

Example 1: A square female tee at 1/2.



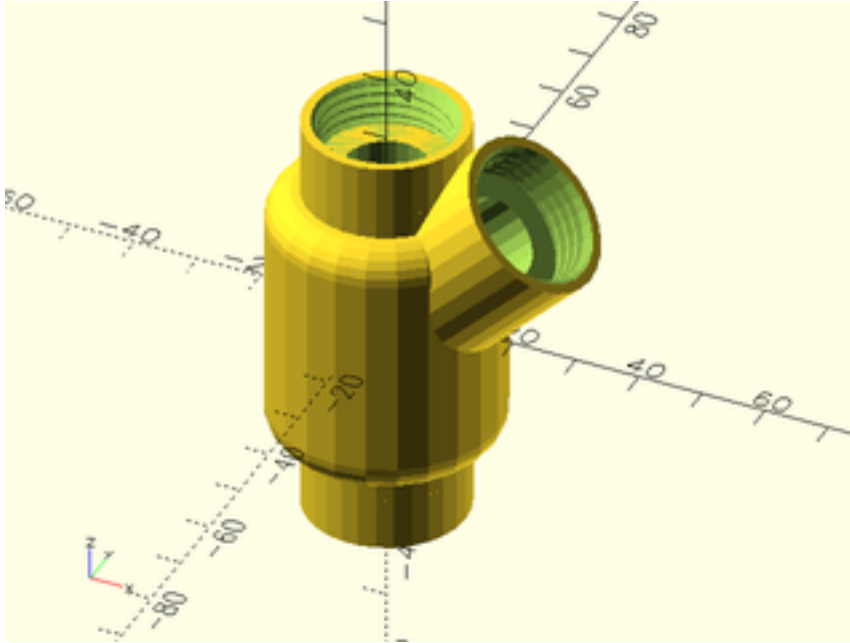
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_tee();
```

Example 2: A reducing tee, 3/4 run with a 1/2 branch.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_tee(size_run=3/4, size_branch=1/2);
```

Example 3: A 45-degree lateral.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_tee(branch_angle=45);
```

Module: `bsppfit_cross()`

Synopsis: A BSPP cross --- a straight run with two opposite branches. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_tee()`, `bsppfit_manifold()`

Usage:

- `bsppfit_cross(size_run, [size_branch=], [gender_run=], [branch_angle=], ...) [ATTACHMENTS];`

Description:

A BSPP cross: a straight run (two collinear ports on -Z and +Z) with two opposite branches in the X-Z plane, each leaning by `branch_angle` (default 90, a square cross with branches along +X and -X). It is the two-branch case of the tee: run and the shared branches may differ in size and gender, and it delegates straight to the super-class. Both branches share `size_branch`, `gender_branch` and the branch overrides; for independent branches use `bsppfit_manifold()` or `bsppfit()` directly. The module is attachable.

Arguments:

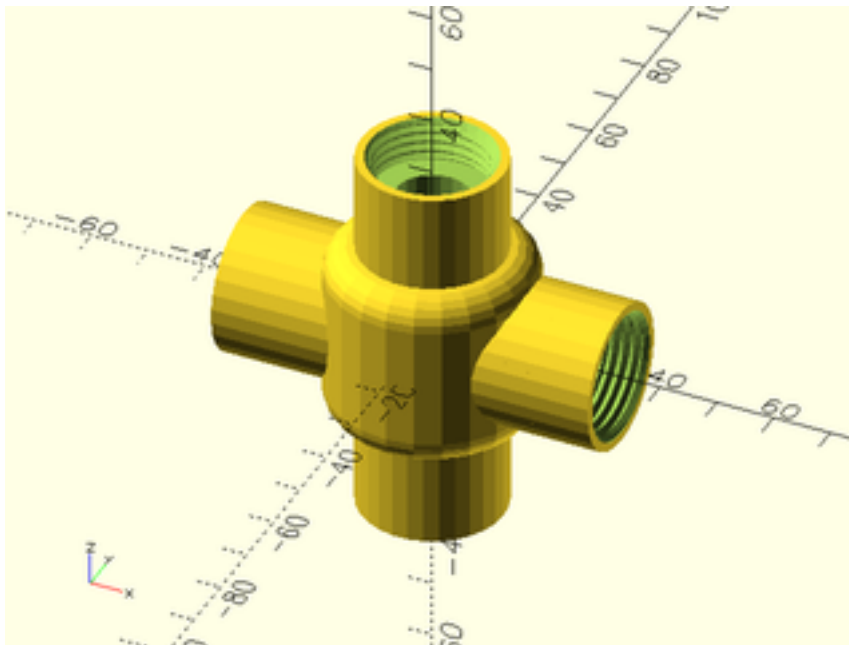
By Position	What it does
<code>size_run</code>	Nominal BSPP size of the run. Default: 1/2
<code>size_branch</code>	Nominal BSPP size of both branches. Default: 1/2

By Name	What it does
<code>gender_run</code>	Gender of both run ports, "M" or "F". Default: "F"

By Name	What it does
<code>gender_branch</code>	Gender of both branches, "M" or "F". Default: "F"
<code>branch_angle</code>	Branch lean from +Z in degrees, 0..180. Default: 90
<code>arm_length_run</code>	Length of each run arm. Default: undef (size-derived)
<code>arm_length_branch</code>	Length of each branch arm. Default: undef (size-derived)
<code>arm_od_run</code>	Outer diameter of the run arms. Default: undef (size-derived)
<code>arm_od_branch</code>	Outer diameter of the branch arms. Default: undef (size-derived)
<code>bore_run</code>	Through-bore diameter of the run. Default: undef (size default)
<code>bore_branch</code>	Through-bore diameter of the branches. Default: undef (size default)
<code>thread_length_run</code>	Thread length of the run ports. Default: undef (derived)
<code>thread_length_branch</code>	Thread length of the branches. Default: undef (derived)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

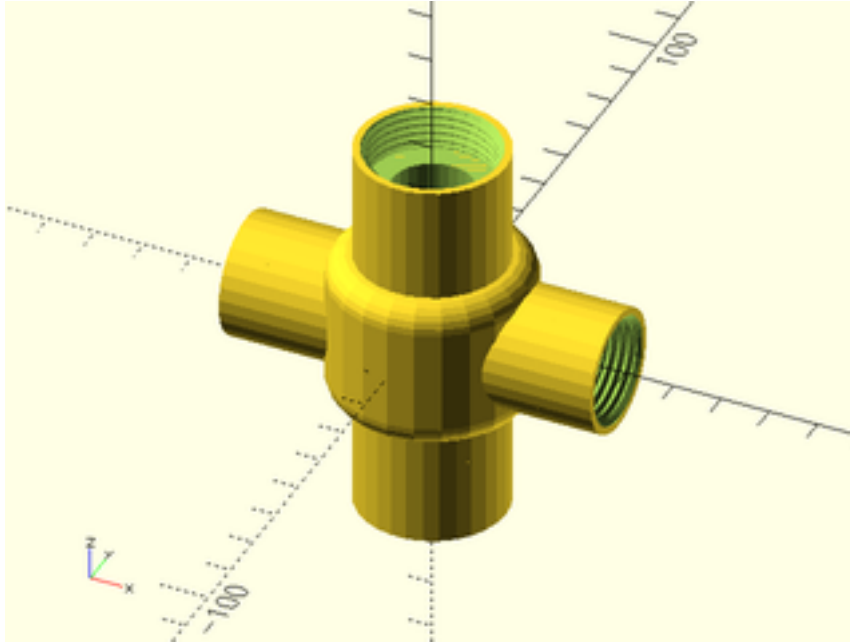
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef anchor | Translate so the anchor point is at the origin. Default: CENTER spin | Rotate this many degrees around the Z axis, after anchoring. Default: 0 orient | Vector to rotate the top toward, after spin. Default: UP

Example 1: A square female cross at 1/2.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cross();
```

Example 2: A reducing cross, 3/4 run with 1/2 branches.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_cross(size_run=3/4, size_branch=1/2);
```

Module: `bsppfit_manifold()`

Synopsis: A BSPP manifold --- a run carrying a family of branches. [Geom]

Topics: Parts, Threading, Pipe Fittings

See Also: `bsppfit_tee()`, `bsppfit_cross()`

Usage:

- `bsppfit_manifold(size_run, [branch_count=], [pitch=], [branch_sides=], ...) [ATTACHMENTS];`

Description:

A BSPP manifold: a straight run carrying a family of branches spread along it --- the most general run-and-branch fitting, with the tee and cross as its one-branch cases. `branch_count` branches sit at `pitch` spacing centred on the run; `branch_sides` "one" puts them on the +X side only, "both" mirrors a family on -X. Each side's lean is set by `top_angle` and `bottom_angle` (each defaulting to `branch_angle`); each side's gender and branch arm length may likewise be overridden (`top_gender_branch`/`bottom_gender_branch`, `top_arm_length_branch`/`bottom_arm_length_branch`), each defaulting to the scalar `gender_branch`/`arm_length_branch`; the two run ends may likewise differ in gender (`left_gender_run`, `right_gender_run`, defaulting to `gender_run`). The run auto-extends to clear leaning branches; either run end (`end_a`, `end_b`) may instead be "closed" (a dead-header) or "point" (that side of the run collapses to zero length, so the nearest branch(es) meet the hub directly --- the catalogue run-as-point Y-piece; at most one end may be "point"). Run and branch may differ in size and gender. The module is attachable.

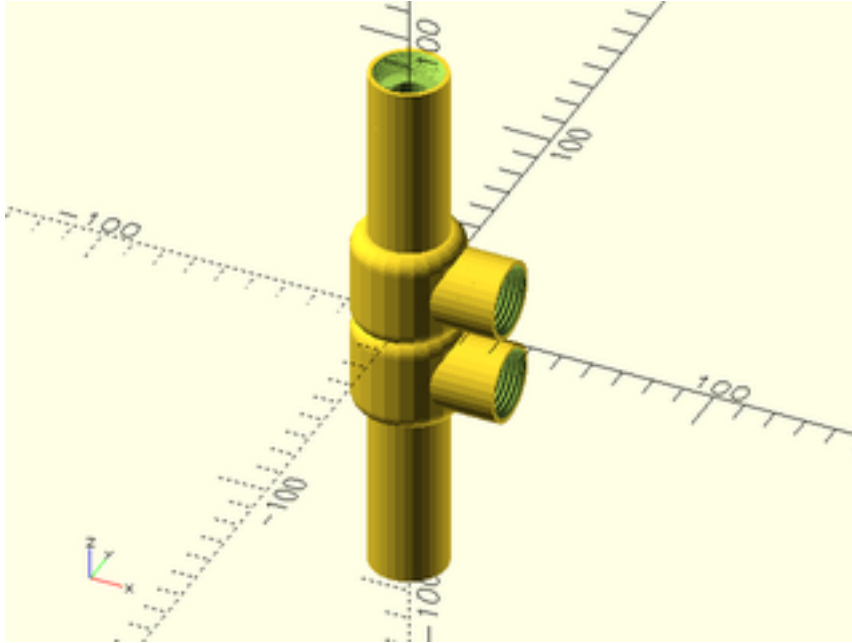
Arguments:

By Position	What it does
<code>size_run</code>	Nominal BSPP size of the run. Default: 1/2
<code>size_branch</code>	Nominal BSPP size of the branches. Default: 1/2

By Name	What it does
<code>gender_run</code>	Gender of the run ports, "M" or "F". Default: "F"
<code>gender_branch</code>	Gender of the branches, "M" or "F". Default: "F"
<code>left_gender_run</code>	Per-end override for the -Z (<code>end_a</code>) run gender. Default: undef (<code>gender_run</code>)
<code>right_gender_run</code>	Per-end override for the +Z (<code>end_b</code>) run gender. Default: undef (<code>gender_run</code>)
<code>top_gender_branch</code>	Per-side override for the +X branches' gender. Default: undef (<code>gender_branch</code>)
<code>bottom_gender_branch</code>	Per-side override for the -X branches' gender. Default: undef (<code>gender_branch</code>)
<code>branch_count</code>	Number of branches per side (≥ 1). Default: 2
<code>branch_sides</code>	"one" (+X only) or "both" (mirror on -X). Default: "one"
<code>pitch</code>	Branch spacing along the run. Default: 30
<code>branch_angle</code>	Default branch lean from +Z in degrees. Default: 90
<code>top_angle</code>	Lean of the +X branches. Default: undef (<code>branch_angle</code>)
<code>bottom_angle</code>	Lean of the -X branches. Default: undef (<code>branch_angle</code>)
<code>end_a</code>	-Z run end: "open", "closed" (header), or "point" (run-as-point). Default: "open"
<code>end_b</code>	+Z run end: "open", "closed", or "point". Default: "open"
<code>arm_length_run</code>	Length of each run arm. Default: undef (size-derived)
<code>arm_length_branch</code>	Length of each branch arm. Default: undef (size-derived)
<code>top_arm_length_branch</code>	Per-side override for the +X branch arm length. Default: undef (<code>arm_length_branch</code>)
<code>bottom_arm_length_branch</code>	Per-side override for the -X branch arm length. Default: undef (<code>arm_length_branch</code>)
<code>arm_od_run</code>	Outer diameter of the run arms. Default: undef (size-derived)
<code>arm_od_branch</code>	Outer diameter of the branch arms. Default: undef (size-derived)
<code>bore_run</code>	Through-bore diameter of the run. Default: undef (size default)
<code>bore_branch</code>	Through-bore diameter of the branches. Default: undef (size default)
<code>thread_length_run</code>	Thread length of the run ports. Default: undef (derived)
<code>thread_length_branch</code>	Thread length of the branches. Default: undef (derived)
<code>manifold_safe</code>	If true, nudge coincident faces apart so the result stays manifold. Default: true
<code>tol_thread</code>	Thread-fit tolerance passed to the thread former. Default: BSPPFIT_TOL_THREAD
<code>undersize</code>	Amount to decrease the male thread diameters, in mm, without changing

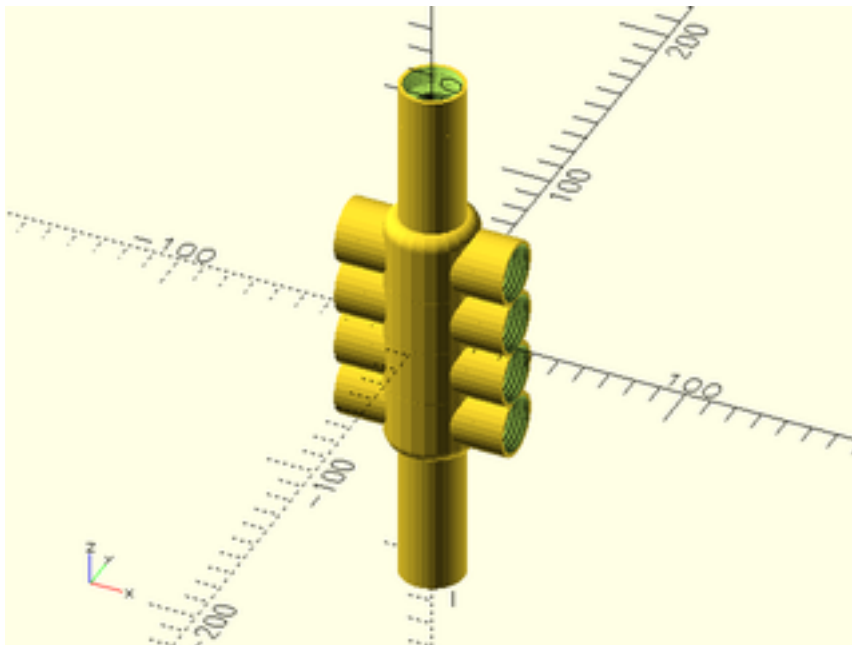
pitch or thread depth; negative values enlarge them. Female threads are unaffected (see `tol_thread`). Default: BSPPFIT_UNDERSIZE cutaway | If set to a side (e.g. FRONT), remove that half for inspection. Default: undef `anchor` | Translate so the anchor point is at the origin. Default: CENTER `spin` | Rotate this many degrees around the Z axis, after anchoring. Default: 0 `orient` | Vector to rotate the top toward, after spin. Default: UP

Example 1: A two-branch manifold on one side.



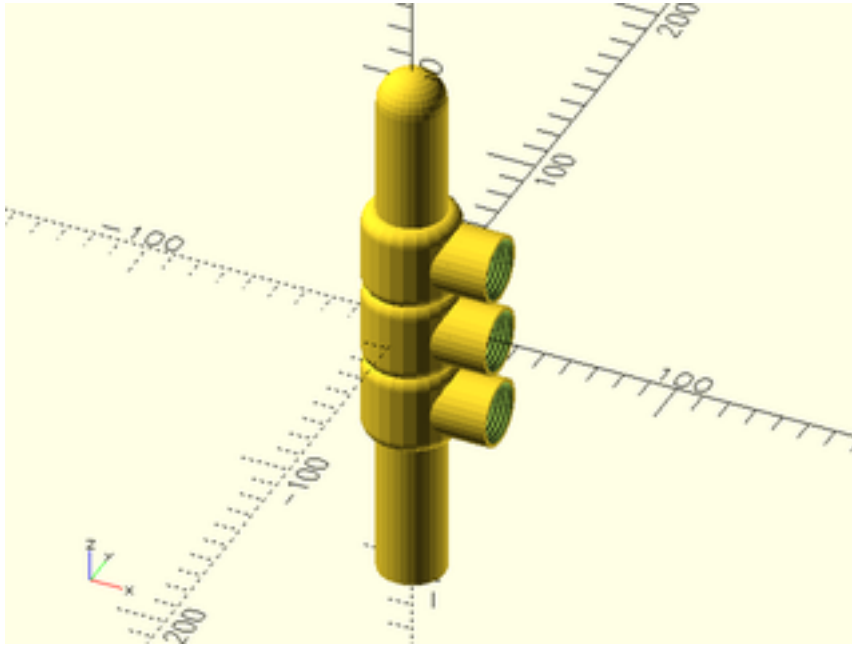
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold();
```

Example 2: Four branches, both sides, closer pitch.



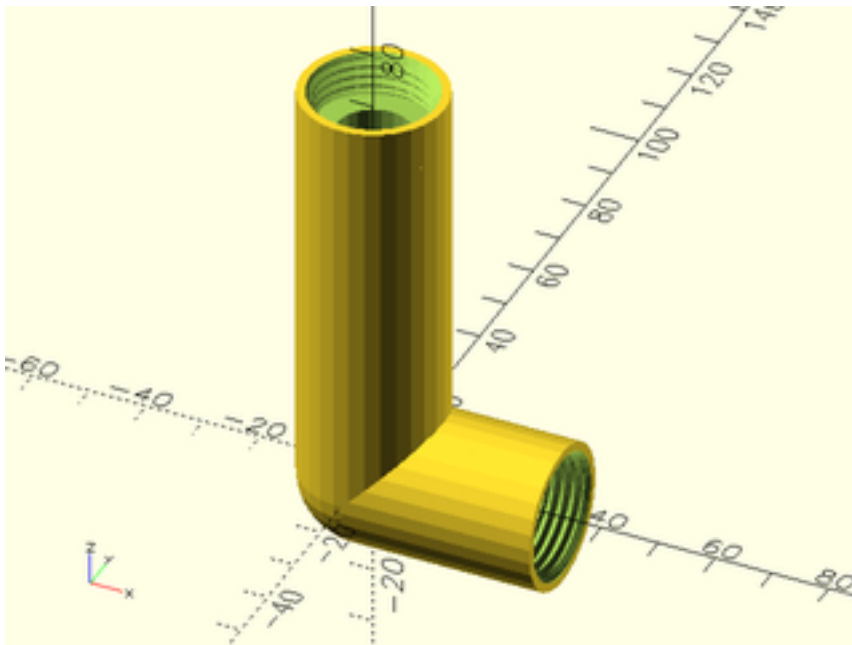
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold(branch_count=4, branch_sides="both", pitch=22);
```

Example 3: A dead-header: one run end closed.



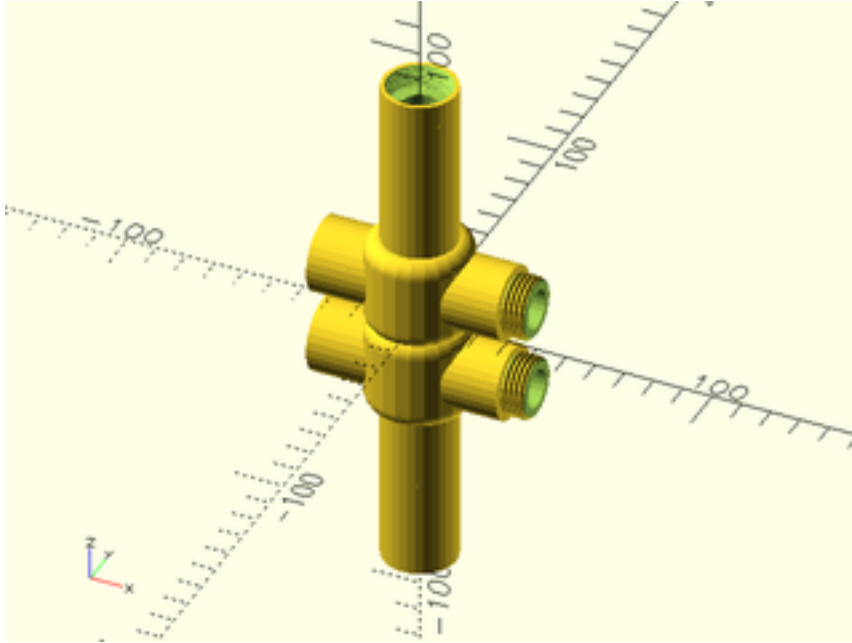
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold(branch_count=3, end_b="closed");
```

Example 4: A run-as-point Y-piece: one run end reduced to a point.



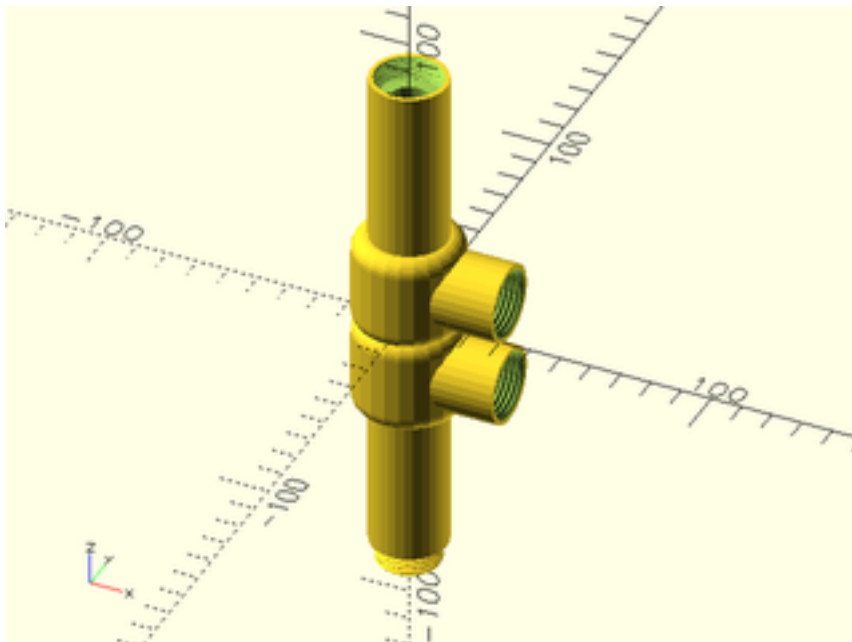
```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold(branch_count=1, end_a="point");
```

Example 5: Mixed branch genders and a per-side arm length.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold(branch_count=2, branch_sides="both",
                 top_gender_branch="M", bottom_gender_branch="F",
                 top_arm_length_branch=40);
```

Example 6: Mixed run-end genders: a male inlet, a female outlet.



```
include <BOSL2/std.scad>
include <BOSL2/threading.scad>
use <bsppfit.scad>
bsppfit_manifold(branch_count=2, left_gender_run="M", right_gender_run="F");
```
