

tikzphysics

TikZ-native physics diagrams

Version 1.6.0

Vaibhav Blayer

vaibhavblayer@gmail.com

2026-09-27

Abstract

`tikzphysics` extends TikZ with reusable shapes and geometric anchors for classical-physics diagrams. It provides continuous contact surfaces, straight and circular ramps, inclined planes, blocks, springs, pulleys, differential elements, shell and slice constructions, native fluid vessels, and optical components. Mechanics primitives include particles, disks, supports, vectors, and pendulums. Diagrams use ordinary TikZ nodes, paths, and named pics.

In one line: `\node[wedge, wedge angle=30] (W) {};`

Contents

1	Start here	4
1.1	Your first diagram in five minutes	4
1.2	How to explore this manual	4
1.3	Loading	4
1.4	The design model	5
1.5	Set defaults once	6
1.6	Dimensions	6
1.7	Complete copy-paste examples	7
2	Contact surfaces and wedges	10
2.1	Bent platforms	10
2.2	Wedges	14
2.3	Ground, ceiling, and walls	18
3	Straight and curved ramps	19
3.1	Straight ramp geometry	19
3.2	Circular curved ramp geometry	20
3.3	Surface, tangent, and normal anchors	20
3.4	Styling and debug use	22
4	Blocks, springs, and pulleys	23
4.1	Complete spring-block-pulley system	23
4.2	Native rope connections	24
4.3	Spring nodes and paths	24
4.4	Blocks, pulleys, and tangent strings	25
4.5	String-routing variants	25
4.6	Complete pulley systems	27
4.7	Bodies, forces and named assemblies	28

5	Circular differential elements	29
5.1	General polar element	29
5.2	Sector, ring, and unwrapped strip	29
5.3	Solid, shell, slice, and sheet constructions	31
5.4	Labels, anchors, and resolved values	32
6	Fluid mechanics	32
6.1	Native nodes and patterns	33
6.2	Composition and native styling	33
6.3	Diagram families	34
6.4	Labels, hooks, and geometry limits	34
7	Optical components	34
7.1	Concave and convex mirrors	35
7.2	Convex and concave lenses	36
7.3	Slab	36
7.4	Prism	36
7.5	Short numeric anchors	37
7.6	Complete copy-paste ray diagram	37
8	Shared tools	38
8.1	Percentage anchors: a beginner's guide	38
8.2	Combining x and y coordinates	41
8.3	Explore keys and anchors inside TikZ	41
8.4	Troubleshooting and geometry limits	43
9	Installation and development	44
10	Complete feature reference	44
10.1	platform-left (node)	44
10.2	platform-right (node)	45
10.3	platform (node)	45
10.4	wedge (node)	46
10.5	ground (node)	46
10.6	ceiling (node)	47
10.7	wall-left (node)	47
10.8	wall-right (node)	47
10.9	ramp (node)	48
10.10	curved-ramp (node)	48
10.11	block (node)	48
10.12	pulley (node)	49
10.13	particle (node)	49
10.14	disk (node)	49
10.15	ring (node)	50
10.16	spring (node and path)	50
10.17	rope (path)	50
10.18	force (path)	51
10.19	velocity (path)	51
10.20	acceleration (path)	51
10.21	torque (path)	51
10.22	rod (path)	51
10.23	pin-support (pic)	51
10.24	roller-support (pic)	51
10.25	pendulum (pic)	52

10.26 polar element (node)	52
10.27 differential sector (node)	52
10.28 differential ring (node)	53
10.29 unwrapped ring (node)	53
10.30 spherical shell (node)	53
10.31 hollow sphere (node)	54
10.32 rectangular element (node)	54
10.33 rectangular strip (node)	54
10.34 rectangular sheet (node)	55
10.35 differential ring diagram (pic)	55
10.36 sphere shell diagram (pic)	55
10.37 sphere slice diagram (pic)	55
10.38 hollow sphere diagram (pic)	56
10.39 cylinder shell diagram (pic)	56
10.40 cylinder slice diagram (pic)	56
10.41 cone slice diagram (pic)	57
10.42 sheet element diagram (pic)	57
10.43 u tube diagram (pic)	57
10.44 gas manometer (pic)	57
10.45 hydraulic press (pic)	58
10.46 fluid cylinder diagram (pic)	58
10.47 fluid tank diagram (pic)	58
10.48 pressure element diagram (pic)	58
10.49 rotating tube (pic)	59
10.50 rotating fluid diagram (pic)	59
10.51 buoyancy (pic)	59
10.52 flow tube diagram (pic)	59
10.53 meniscus diagram (pic)	60
10.54 capillary (pic)	60
10.55 surface tension ring (pic)	60
10.56 liquid ring diagram (pic)	60
10.57 fluid tank (node)	61
10.58 meniscus (node)	61
10.59 rotating fluid (node)	61
10.60 fluid cylinder (node)	61
10.61 pressure element (node)	62
10.62 flow tube (node)	62
10.63 liquid ring (node)	62
10.64 u tube (node)	63
10.65 concave-mirror (node)	63
10.66 convex-mirror (node)	63
10.67 convex-lens (node)	63
10.68 concave-lens (node)	64
10.69 slab (node)	64
10.70 prism (node)	64

11 License	64
-------------------	-----------

1 Start here

1.1 Your first diagram in five minutes

Install the package normally, or upload the generated single-file output `/overleaf/tikzphysics.sty` beside an Overleaf document. Compile this complete document with pdfLaTeX. No shell escape is needed to use the package.

```
\documentclass[tikz,border=5mm]{standalone}
\usepackage{tikzphysics}
\begin{document}
\begin{tikzpicture}
  \node[block] (B) at (3,0) {$m$};
  \draw[spring] (0,0) -- node[above] {$k$} (B.west);
\end{tikzpicture}
\end{document}
```



Read this as: create a block, name it B, and connect a spring to its west anchor. Change (3,0) to move the block; change `minimum width` to resize it. The spring follows its endpoints.

1.2 How to explore this manual

Your next task	Where to go
Choose defaults for a whole worksheet	Section 1.5
Build the spring–block–pulley example	Section 4.1
Find anchors while drawing	Section 8.3
Draw supports, vectors, or pendulums	Section 4.7
Draw a polar area or differential ring	Section 5
Look up every feature's keys and anchors	Section 10
Troubleshoot placement or transformations	Section 8.4

The examples directory contains complete documents. Its `README.md` provides a learning route and explains what each example demonstrates.

1.3 Loading

Load every module with the package wrapper:

```
\usepackage{tikzphysics}
```

For smaller documents, load only the required TikZ libraries:

```
\usepackage{tikz}
\usetikzlibrary{tikzphysics.surface, tikzphysics.mechanics}
```

The available libraries are `tikzphysics.surface`, `tikzphysics.ramps`, `tikzphysics.mechanics`, `tikzphysics.elements`, `tikzphysics.fluids`, `tikzphysics.optics`, and `tikzphysics.core`. Each feature library loads `tikzphysics.core` automatically.

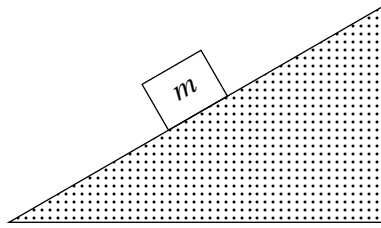
1.4 The design model

1. Choose a node style such as `wedge`, `block`, `convex-lens`, `concave-lens`, `slab`, or `prism`.
2. Use familiar TikZ sizing keys where they apply, and `physics ...` keys for shape-specific geometry or as unit-aware conveniences.
3. Give each node a name and compose the diagram through its anchors; draw a spring directly between any two such coordinates.

Short style names are the normal user interface. The longer forms remain available when another package or the surrounding document already defines a generic TikZ style such as `block`.

Short style	Collision-safe style	Object
<code>platform</code>	<code>physicsplatform-both</code>	General platform with two configurable walls
<code>platform-left</code> / <code>platform-right</code> / <code>platform-both</code>	<code>physicsplatform-left</code> / <code>physicsplatform-right</code> / <code>physicsplatform-both</code>	Explicit platform variants
<code>platform-left-up</code> / <code>platform-right-up</code>	<code>physicsplatform-left-up</code> / <code>physicsplatform-right-up</code>	One-wall presets with wall angle=90
<code>wedge</code>	<code>physicswedge</code>	Inclined plane
<code>ramp</code> / <code>ramp-left</code>	<code>physicsramp</code>	Continuous wall-floor-linear-incline body
<code>curved-ramp</code> / <code>curved-ramp-left</code>	<code>physicscurvedramp</code>	Circular contact ramp with a filled body
<code>ground</code> / <code>ceiling</code>	<code>physicsground</code> / <code>physicsceiling</code>	Horizontal contact strips
<code>wall-left</code> / <code>wall-right</code>	<code>physicswall-left</code> / <code>physicswall-right</code>	Vertical contact strips
<code>block</code> / <code>pulley</code>	<code>physicsblock</code> / <code>physicspulley</code>	Mechanics nodes
<code>spring</code>	<code>physics spring</code> / <code>physicsspring</code>	Positioned node or coil decoration for paths
<code>polar element</code>	<code>physicspolarelement</code>	General annular-sector differential element
<code>differential sector</code>	<code>physicsdifferentialsector</code>	Zero-inner-radius sector preset
<code>differential ring</code>	<code>physicsdifferentialring</code>	Full thin-annulus preset
<code>unwrapped ring</code>	<code>physicsunwrappedring</code>	Equivalent rectangular strip
<code>concave-mirror</code> / <code>convex-mirror</code>	<code>physicsconcavemirror</code> / <code>physicsconvexmirror</code>	Curved mirrors
<code>convex-lens</code>	<code>physicsconvexlens</code>	Symmetric biconvex lens
<code>concave-lens</code>	<code>physicsconcavelens</code>	Symmetric biconcave lens
<code>slab</code>	<code>physicsslab</code>	Parallel-sided rectangular slab
<code>prism</code>	<code>physicsprism</code>	Isosceles triangular prism

```
\begin{tikzpicture}
  \node[wedge, pulley edge, wedge width=5.6, wedge angle=30] (W) at (0,0) {};
  \node[block, minimum width=0.9cm, minimum height=0.7cm,
    rotate=30, anchor=south] at (W.slope-mid) {$m$};
\end{tikzpicture}
```



1.5 Set defaults once

The default block is a 1 cm square and the default pulley diameter is 1 cm. Simple names are the public interface: `spring`, `block`, `pulley`, and `wedge`. No slash-prefixed style is required. Each node or path style offers an `every <name>` hook:

```
\tikzset{
  every block/.style={minimum width=1cm,minimum height=1cm,fill=white},
  every spring/.style={pre length=3mm,post length=3mm,amplitude=2mm},
  every pulley/.style={minimum size=1cm}
}
\node[block] (A) {};
\node[block,minimum width=2cm] (B) at (3,0) {};
\draw[spring] (A.east) -- (B.west);
```

For node and path styles, the order is built-in settings, the shared hook, the feature hook, then later options on your node or path. This means a local `minimum width=2cm` overrides `every block` when placed after `block`. A scope contains its own settings without changing later pictures. A bare `minimum width` set at picture level may still be replaced by the object's built-in dimensions; use `every block` for object-wide sizing.

`every physics` object applies to package node styles; `every physics connection` applies to package path styles. `every platform` covers all platform variants; the other node/path hooks use the short feature name. The collision-safe hook `every physics block` is called by the initial definition of `every block`; redefining the short hook replaces that forwarding style. Do not put `block` inside `every block`, since that would call the hook recursively.

You can also use `spring/.append style={...}`. Hooks are a convenient, explicit place to keep preferences while retaining the short drawing syntax. Built-in defaults are not the same as a key's missing-value default: `show anchors` means `show anchors=true`, whereas geometric keys such as `wedge angle` require a value. An omitted geometry key keeps its inherited or built-in setting.

1.6 Dimensions

TikZ's native sizing keys are fully supported and are often the most familiar choice. Use explicit units with them, for example `minimum width=1.2cm`, `minimum height=8mm`, or `minimum size=0.7cm`. Package convenience keys may be used instead. They accept explicit dimensions and also interpret a bare value as centimetres, so `block width=1.2` and `minimum width=1.2cm` produce the same block size requirement.

Native TikZ (preferred)	Package convenience	Applies to
minimum width / height	block width / height	block
minimum size	pulley diameter	pulley
minimum width / height	platform width / depth	Bent platforms
minimum width / height	ground width / depth	ground
minimum width / height	ceiling width / depth	ceiling
minimum width / height	wall thickness / height	Left and right walls
minimum width	wedge width	wedge width only
minimum width	--	ramp overall horizontal width
minimum width / height	slab width / height	slab
minimum width / height	prism width / height	prism

The alternatives in each row set the same underlying TikZ size requirement. If both are supplied, normal TikZ option ordering applies: the last value wins. Put the shape style first, then the desired sizing keys, for example `block, minimum width=1.2cm`. These are minimum dimensions; a node may grow to contain wider or taller text.

Shape-specific keys remain necessary when they control more than a native node size. In particular, use `wedge height` rather than `minimum height` when the height should determine wedge geometry. Keys such as `wall angle`, `strip width`, `convex lens radius`, and `prism apex angle` have no native TikZ sizing equivalent. All package size keys require positive values. Each concise wedge key also has a collision-safe form prefixed by `physics`, as listed in the wedge section.

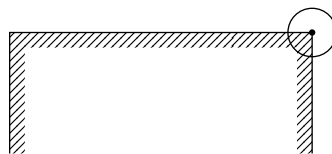
1.7 Complete copy-paste examples

Each example below includes the document class, package loading, document environment, and complete TikZ picture. The first shows the basic composition pattern: name a node, then place another node at one of its anchors.

```
\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}

\begin{document}
\begin{tikzpicture}
  \node[platform, minimum width=5cm, minimum height=2cm]
    (platform) at (0,0) {};

  \node[pulley, minimum size=8mm]
    (pulley) at (platform.north east) {};
\end{tikzpicture}
\end{document}
```



The next complete document composes an inclined plane, two blocks, a pulley, and one exactly tangent string.

```

\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}

\begin{document}
\begin{tikzpicture}
  \node[ground, ground width=8.6cm, ground depth=3mm,
    anchor=top-left] (ground) at (-0.6,0) {};
  \node[wedge, pulley edge, wedge width=7cm, wedge angle=30]
    (plane) at (0,0) {};

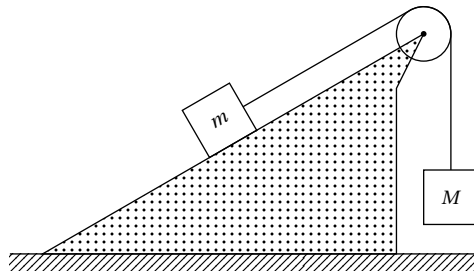
  \node[block, minimum width=1cm, minimum height=1cm,
    rotate=30, anchor=south]
    (surface-block) at (plane.slope-mid) {$m$};

  \node[pulley] (pulley) at (plane.pulley-center) {};

  \node[block, anchor=north]
    (load) at ($(pulley.east)+(0,-2.5cm)$) {$M$};

  \physicsstringoverpulley{surface-block.east}{pulley}{load.north}
\end{tikzpicture}
\end{document}

```



The 7 cm, 30-degree wedge resolves to a height of about 4.04 cm. Placing the hanging block's north anchor 2.5 cm below the pulley centre leaves the full 1 cm block about 5.4 mm above the ground.

Wedge key variants

This complete document demonstrates angle-driven and height-driven geometry, both width interfaces, and every value of `wedge right angle at`.

```

\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}

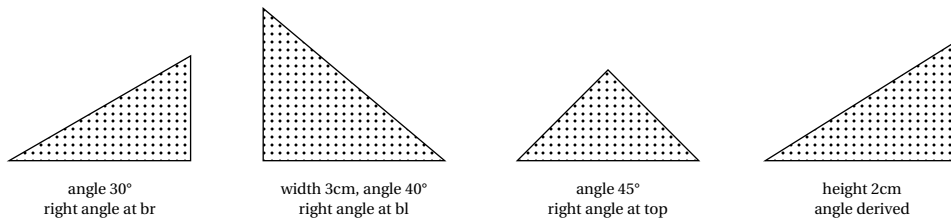
\begin{document}
\begin{tikzpicture}[font=\small]
  \node[wedge, minimum width=3cm, wedge angle=30,
        anchor=base-mid]
    (angle-br) at (0,0) {};
  \node[align=center] at (0,-0.65)
    {angle 30\textdegree\right angle at br};

  \node[wedge, wedge width=3, wedge angle=40,
        wedge right angle at=bl, anchor=base-mid]
    (angle-bl) at (4.2,0) {};
  \node[align=center] at (4.2,-0.65)
    {width 3cm, angle 40\textdegree\right angle at bl};

  \node[wedge, minimum width=3cm, wedge angle=45,
        wedge right angle at=top, anchor=base-mid]
    (angle-top) at (8.4,0) {};
  \node[align=center] at (8.4,-0.65)
    {angle 45\textdegree\right angle at top};

  \node[wedge, minimum width=3.2cm, wedge height=2cm,
        anchor=base-mid]
    (height-driven) at (12.6,0) {};
  \node[align=center] at (12.6,-0.65)
    {height 2cm\angle derived};
\end{tikzpicture}
\end{document}

```



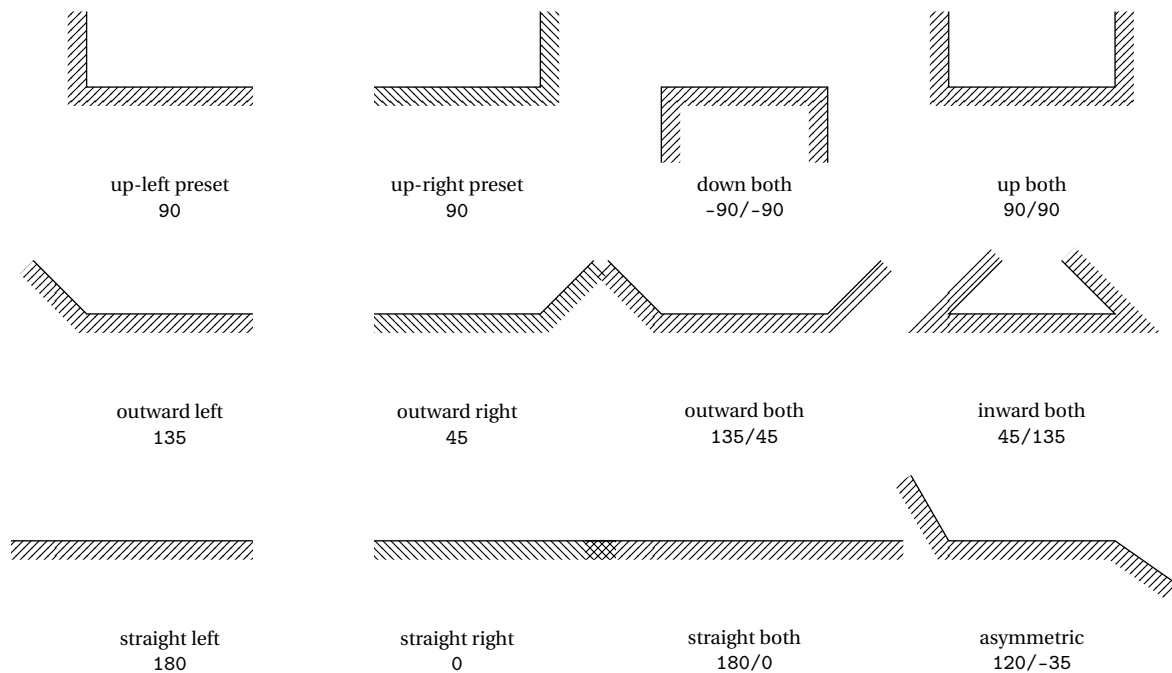
2 Contact surfaces and wedges

2.1 Bent platforms

The three underlying platform shapes share a horizontal floor strip. The left and right variants extend that strip through one bent wall; the general platform style uses the two-wall shape and is equivalent to the explicit `platform-both` style. Each node is rendered as one continuous platform, without a hinge mark or seam. Wall direction is continuous rather than limited to presets, so the gallery below covers every distinct configuration class; intermediate angles use the same keys.

Style	Configuration
<code>platform</code>	General two-wall platform; equivalent to <code>platform-both</code> .
<code>platform-left</code>	One wall at the left floor corner.
<code>platform-right</code>	One wall at the right floor corner.
<code>platform-left-up</code>	Left-wall preset with <code>wall angle=90</code> .
<code>platform-right-up</code>	Right-wall preset with <code>wall angle=90</code> .
<code>platform-both</code>	Two walls; set both with <code>wall angle</code> , or set them independently with <code>left wall angle</code> and <code>right wall angle</code> .

Variant gallery



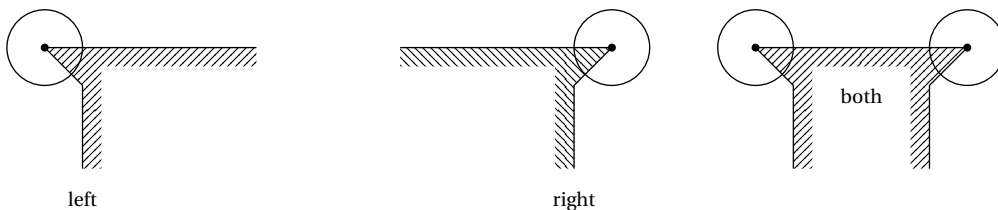
Concise key	Collision-safe alias	Meaning and default
minimum width / platform width	physics platform width	Overall floor width; default 5cm.
minimum height / platform depth	physics platform depth	Wall length and nominal vertical extent; default 2cm.
strip width	physics strip width	Thickness of the floor and walls; default 0.25cm.
wall angle	physics wall angle	Direction of every wall, measured counter-clockwise from the positive x-axis; default -90.
left wall angle	physics left wall angle	Left-wall override for platform-both.
right wall angle	physics right wall angle	Right-wall override for platform-both.
wall inset	physics wall inset	Horizontal distance from each projecting floor tip to its wall root; default 0cm.
wall drop	physics wall drop	Downward distance from each projecting floor tip to its wall root; default 0cm.
left/right wall inset	physics left/right wall inset	Independent inset overrides for platform-both.
left/right wall drop	physics left/right wall drop	Independent drop overrides for platform-both.

Continuous pulley edges

The `pulley edge` preset adds the tapered corner found in many textbook pulley diagrams. It sets `wall inset=0.5cm` and `wall drop=0.5cm`, half the default pulley diameter in each direction. The floor, diagonal transition, and wall remain one closed, patterned path. There is no separately overlaid bracket. An upward wall meets the floor at its sharp corner; keep its wall inset and wall drop at zero. Inset and drop are for walls extending below the floor.

```
\node[platform-right, pulley edge, platform width=5] (S) {};
\node[pulley] (P) at (S.pulley-center) {};

\node[platform, pulley edges] (Both) {};
\node[pulley] at (Both.left-pulley-center) {};
\node[pulley] at (Both.right-pulley-center) {};
```



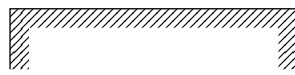
On a one-wall platform, `pulley-center` marks the projecting tip. The `wall-root` anchor marks the end of the transition; `transition-0..100` addresses points along it, with `transition-50` at its midpoint. For `platform-both`, prefix these anchors with `left-` or `right-`. The styles `left pulley edge` and `right pulley edge` enable one side independently. Shared and side-specific `inset/drop` keys accept bare centimetre values or explicit TeX dimensions. Both values default to zero, so existing platform drawings retain their original geometry. The length set by `platform depth` remains the wall length from the wall root; the transition drop adds to the complete vertical extent.

Use `show anchors` with the `transition` family to inspect points on the diagonal. The complete in-picture reference card also lists every new key and anchor. See `examples/debug-pulley-edge.tex` for a

standalone example.

Key-by-key variants

Each public platform key can be varied independently and combined with any configuration from the preceding gallery. These examples isolate the original sizing and direction controls; the pulley-edge controls are shown above.



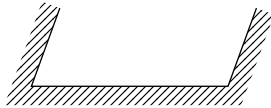
wide floor
platform width=3.8



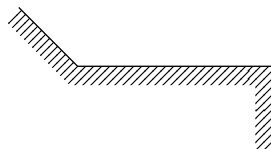
deep walls
platform depth=1.5



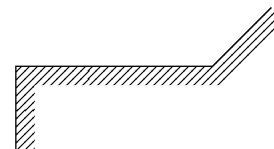
thick material
strip width=0.5



shared direction
wall angle=70

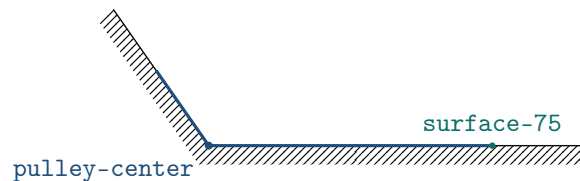


left override
left wall angle=135



right override
right wall angle=45

With zero inset and drop, the floor and wall contact lines meet at pulley-center. With either transition dimension set, that point remains the projecting floor tip and the wall begins at wall-root. The complete body is still one closed polygon with exact inner miters. The two-wall style exposes left-pulley-center and right-pulley-center; these lie on the same continuous geometry. Angles may vary continuously; only a left wall at 0° and a right wall at 180° (modulo 360°) are excluded because they retrace the floor and have no finite miter.



Platform anchors

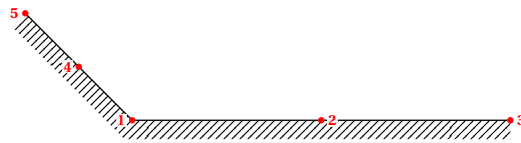
Part	Available anchors
Floor contact	surface-0..100, left to right; surface is its midpoint.
Other floor edges	bottom-0..100 left to right; left-0..100 and right-0..100 bottom to top.
Wall edges	wall-surface-0..100 and wall-back-0..100 root to tip; wall-base-0..100 and wall-tip-0..100 outer to inner.
Transition	transition-0..100 from pulley-center to wall-root.
Inside	floor-center and wall-center.
Two-wall form	Prefix wall and transition families with left- or right-; use left-pulley-center and right-pulley-center.

Every boundary family accepts integer percentages from 0 through 100; for example, surface-0, surface-50, and surface-100 mark the floor contact face's start, midpoint, and end.

To see the floor without a crowded wall legend, use `show anchors` and select its four edge families at 0, 50, and 100:

```
physics debug/anchor families={surface,bottom,left,right},
physics debug/anchor samples={0,50,100}
```

Inspect a wall separately with the transition, wall-surface, wall-back, wall-base, and wall-tip families. On platform, prefix these five with left- or right-. The complete thirteen-page visual reference is `surface-anchor-coverage.pdf` in `docs/`. Its copy-ready source is `surface-anchor-coverage.tex` in `examples/`.

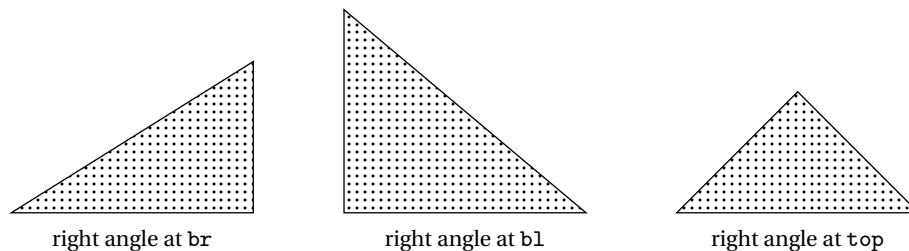


Anchors			
#	Name	#	Name
1	pulley-center	1	surface-0
2	surface-50	3	surface-100
1	wall-surface-0	4	wall-surface-50
5	wall-surface-100		

An upward wall has zero inset and drop, so its visible contact line meets the floor at one corner.

2.2 Wedges

`wedge` creates a right triangle for inclined-plane diagrams. Its material uses TikZ's `dots` pattern by default; a later `pattern=...` option can replace it normally.



Concise key	Collision-safe alias	Meaning and default
wedge width	physics wedge width	Horizontal foot-to-tip span; default 4cm. A pulley edge shortens the base by its inset.
wedge height	physics wedge height	Explicit height when no angle is supplied.
wedge angle	physics wedge angle	Authoritative acute angle; default is 30 degrees for bl/br, and 45 degrees for top.
wedge right angle at wedge top inset	physics wedge right angle at physics wedge top inset	One of br (default), bl, or top. Horizontal clearance behind the fixed pulley tip; default 0cm.
wedge top drop	physics wedge top drop	Vertical distance from the pulley tip to the wall root; default 0cm.

Geometry precedence is `wedge angle`, then `wedge height`, then the mode default. The principal anchors are bl, br, top, base-mid, right-mid, slope-mid, slope-right-mid, and centroid. Numeric edge families are base-0..100, right-0..100, slope-0..100, and transition-0..100. right-mid equals right-50, slope-mid equals surface-50, and mid equals the area centroid. The four-point pulley-edge variant computes that centroid from its visible body. The complete rendered wedge anchor map is in `docs/wedge-anchors.md` and `docs/wedge-anchor-coverage.pdf`.

Pulley edge on a wedge

The shared pulley edge preset keeps the wedge tip fixed and inserts a 5 mm inward transition followed by a 10 mm drop. It therefore accommodates a full default 1 cm hanging block beside the supporting wall.

```
\node[ground, ground width=8.6cm, ground depth=3mm,
      anchor=top-left] (G) at (-0.6,0) {};
\node[wedge, pulley edge, wedge width=7cm, wedge angle=30] (W) {};
\node[pulley] (P) at (W.pulley-center) {};
\node[block, anchor=north] (H)
  at ($(P.east)+(0,-2.5cm)$) {$m_2$};
```

The complete patterned wedge is one closed path. The anchors top and pulley-center coincide at the projecting tip. wall-root marks the start of the vertical side, while transition-mid and transition-0..100 sample the diagonal nose. Direct customisation uses wedge top inset and wedge top drop; both default to zero, preserving the triangular wedge. The inset must remain smaller than the wedge width and the drop smaller than its resolved height. For a nonzero inset, the drop must also place the wall root strictly below the incline.

The preset supports the default right angle at=br mode and mirrors for right angle at=bl. It is unavailable in right angle at=top, which has two sloping sides and no vertical supporting side.

```
\node[wedge, pulley edge, show anchors, show keys,
      physics debug/anchor list={pulley-center,wall-root,transition-mid},
      physics debug/anchor families={transition},
      physics debug/anchor samples={0,50,100}] (W) {};
```

Attach to the usable incline

Prefer surface-0..100 when positioning an object on the contact face. Unlike the older CCW edge names, this family selects the actual incline in all three modes and always runs left to right. In top mode it selects the left face; right-surface-0..100 selects the right face. In the other two modes the right family refers to the same incline.

```
\node[wedge,wedge right angle at=bl,wedge angle=30] (W) {};
\path (W.tangent-before-50) -- (W.tangent-after-50)
      node[midway,sloped,block,anchor=south] (B) {$m$};
\draw[force] (W.surface-25) -- (W.normal-25);
```

The endpoint names are surface-start and surface-end. The following guides are saved per node:

Anchor or key	Meaning
tangent-before-T, tangent-after-T	Two points straddling the contact point.
normal-T	A point outside the wedge, along its normal.
surface guide length	Tangent separation and normal length; initially 0.5cm.
right-tangent-before-T, right-tangent-after-T	Corresponding guides on the right face in top mode.
right-normal-T	Normal guide for that face.

Nonuniform scaling changes perpendicularity; use rigid transformations or uniform scaling for physical normals.

Named geometry values

Every named wedge snapshots its resolved scalar geometry. Query the value owned by node W with

$$\backslash geometryvalue{W}\{property\}$$

The collision-safe command `\tikzphysicsgeometryvalue{W}\{property\}` has the same result. The wedge must be named and its node command must finish before either command is used. Because the result is a fixed literal, it can appear directly inside `rotate=...`, a PGF calculation, or a label.

Property	Meaning
left angle	Interior angle at <code>bl</code> , in degrees
right angle	Interior angle at <code>br</code> , in degrees
top angle	Interior angle at <code>top</code> , in degrees
base direction	Direction from <code>bl</code> to <code>br</code>
left edge direction	Rising direction from <code>bl</code> to <code>top</code>
right edge direction	Rising direction from <code>br</code> to <code>top</code>
slope angle, slope direction, surface angle	Placement rotation for the surface at <code>slope-mid</code>
right slope angle	Placement rotation for the surface at <code>slope-right-mid</code>
base angle, right edge angle	Aliases for <code>base direction</code> and <code>right edge direction</code>
width, height	Resolved TeX dimensions
top inset, top drop	Resolved pulley-edge dimensions
wall root angle	Interior angle at <code>wall-root</code> ; 180 degrees when the transition is absent or collinear
right angle at	The text value <code>bl</code> , <code>br</code> , or <code>top</code>

Angles and directions are returned as numbers without a degree symbol. Directions use degrees counter-clockwise from the wedge's local positive x -axis. The placement rotations for all wedge modes are:

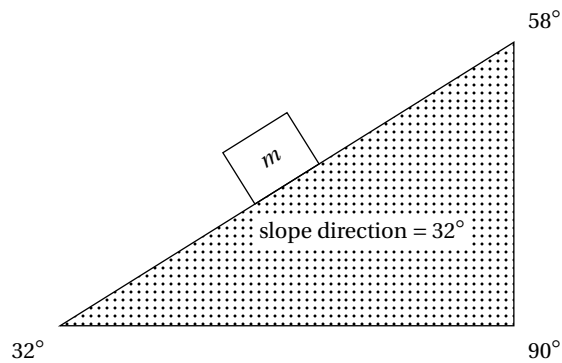
	right angle at	left angle	right angle	top angle	slope angle
br		θ	90	$90 - \theta$	θ
bl		90	θ	$90 - \theta$	$-\theta$
top		θ	$90 - \theta$	90	θ

Here `slope angle` follows the contact surface selected by `slope-mid`; it is therefore the property normally wanted for a block. In `top` mode this is the left face. Use `right slope angle`, equal to $\theta - 90$, with `slope-right-mid` for the right face.

```
\node[wedge, minimum width=6cm, wedge angle=32] (W) at (0,0) {};

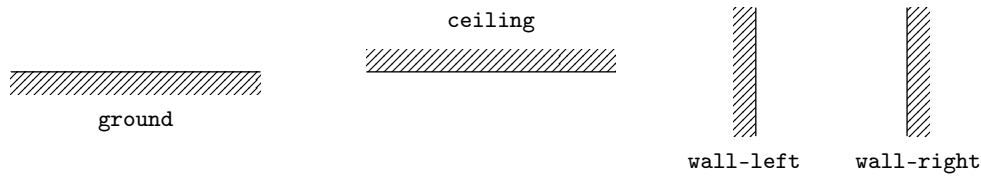
\node[block, minimum width=1cm, minimum height=0.8cm,
  rotate=\geometryvalue{W}{slope angle}, anchor=south]
  (B) at (W.slope-mid) {$m$};

\node at (W.bl)
  {\pgfmathprintnumber{\geometryvalue{W}{left angle}}~\circ};
```



These direction values describe the untransformed wedge. They work directly when the wedge and attached object share the same transformed scope. If the wedge has its own independent rotation or a non-uniform scale, orient the attached object from edge anchors with TikZ's `sloped` placement instead; that follows the final rendered edge. The complete copy-ready source is `examples/wedge-geometry-values.tex`.

2.3 Ground, ceiling, and walls



The sizing keys are grouped by shape:

ground	ground width, ground depth
ceiling	ceiling width, ceiling depth
wall-left, wall-right	wall thickness, wall height

Each shape exposes four corners, center, and surface at the usable face's midpoint. `surface-0..100` covers that face: left to right on ground and ceiling, bottom to top on either wall. The other three edges use their physical names, top, bottom, left, or right, with the same horizontal and vertical directions. Use `-50` for any edge midpoint.

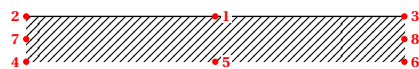
Inspect all four with `show anchors`

Select the contact family and the other three boundaries for each shape. Numbers 0, 50, and 100 show each edge's start, midpoint, and end. Coincident corners share a marker and appear together in the table.

Shape	physics debug/anchor families
ground	{surface,bottom,left,right}
ceiling	{surface,top,left,right}
wall-left	{surface,top,bottom,left}
wall-right	{surface,top,bottom,right}

```
\node[ground,ground width=7cm,show anchors,
  physics debug/anchor list={surface,top-left,top-right},
  physics debug/anchor families={surface,bottom,left,right},
  physics debug/anchor samples={0,50,100}] (G) {};
```

The same `show anchors` overlay is rendered for ground, ceiling, wall-left, wall-right, and every platform in `docs/surface-anchor-coverage.pdf`. The full explanation and copy-ready settings are in `docs/surface-anchors.md`.



Anchors			
#	Name	#	Name
1	surface	2	surface-0
1	surface-50	3	surface-100
4	bottom-0	5	bottom-50
6	bottom-100	4	left-0
7	left-50	2	left-100
6	right-0	8	right-50
3	right-100		

3 Straight and curved ramps

The `tikzphysics.ramps` library provides two comparatively unusual mechanics surfaces as genuine TikZ node shapes. They are not collections of overlaid lines: each ramp body is one closed path that receives the node's fill or pattern, and each usable contact surface is drawn once as a continuous line.



The default direction is right: parameter 0 is the low end and parameter 100 is the high end. Use `ramp direction=left`, or the convenience styles `ramp-left` and `curved-ramp-left`, to mirror the body. The parameter direction still runs from low to high, so algorithms and annotations do not need to reverse their anchor numbers.

3.1 Straight ramp geometry

The straight ramp combines a vertical wall, a horizontal floor, and a linear incline. The floor-to-incline meeting point is a mathematically sharp vertex; no rounding, joint circle, or seam is inserted. `minimum width` is the horizontal distance from the wall contact line to the high end. The `ramp run` occupies the final part of that width, so the flat length is

$$L_{\text{flat}} = \text{minimum width} - \text{ramp run}.$$

When `ramp angle` is supplied it is authoritative and the rise is $h = L_{\text{run}} \tan \theta$. Otherwise an explicit `ramp rise` determines the angle. With neither key, the default angle is 30° . Use `\physicsrampangle{R}{label}` after the node to draw the dashed reference ray, acute angle arc, and label at the exact ramp-foot. The command also works for `ramp-left`; its optional argument accepts ordinary TikZ angle-pic options.

Concise key	Collision-safe alias	Meaning and default
<code>minimum width</code>	--	Overall horizontal width; default 8.6cm.
<code>ramp run</code>	<code>physics ramp run</code>	Horizontal run of the incline; default 2.6cm.
<code>ramp angle</code>	<code>physics ramp angle</code>	Incline angle; default 30 degrees.
<code>ramp rise</code>	<code>physics ramp rise</code>	Explicit rise when no angle is supplied.
<code>ramp wall height</code>	<code>physics ramp wall height</code>	Contact-wall height; default 1.5cm.
<code>ramp wall width</code>	<code>physics ramp wall width</code>	Material behind the wall; default 0.25cm.
<code>ramp depth</code>	<code>physics ramp depth</code>	Material below the baseline; default 0.25cm.
<code>ramp direction</code>	<code>physics ramp direction</code>	right (default) or left.
<code>ramp guide length</code>	<code>physics ramp guide length</code>	Length used by tangent/normal guide anchors; default 0.5cm.

The following is a complete copy-paste document reproducing a wall, spring, block, floor, and rising end as a single ramp node.

```

\documentclass[tikz,border=6mm]{standalone}
\usepackage{tikzphysics}
\begin{document}
\begin{tikzpicture}[>=stealth]
  \node[ramp, minimum width=8.6cm, ramp run=2.6cm,
    ramp rise=1.5cm, ramp wall height=1.5cm] (R) at (0,0) {};

  \node[block, minimum width=1cm, minimum height=0.75cm,
    anchor=south] (B) at (R.floor-mid) {$m$};
  \coordinate (A) at (R.wall-mid |- B.west);
  \draw[spring] (A) -- node[above=3pt] {$k$} (B.west);
  \draw[->,thick] (B.east) -- ++(1.2,0) node[right] {$v$};
  \physicsrampangle{R}{30~\circ}
\end{tikzpicture}
\end{document}

```

3.2 Circular curved ramp geometry

curved-ramp begins with a horizontal floor extending left, then enters a circular arc tangentially at (0,0); it is not an arbitrary Bézier curve. For radius R , sweep angle α , and circular-arc parameter $t = T/100$, the right-facing curve-T point is

$$x(T) = R \sin(t\alpha), \quad y(T) = R(1 - \cos(t\alpha)).$$

Thus a 90-degree sweep ends at (R, R) with a vertical tangent, exactly matching the common quarter-circle ramp. The floor-to-curve transition has a common horizontal tangent, so it contains no corner or visible joint. The body continues horizontally by curved ramp back extension, then closes down to the baseline. The supported sweep range is greater than 0 and at most 90 degrees; this keeps the filled body single-valued and prevents it folding back over itself.

Concise key	Collision-safe alias	Meaning and default
curved ramp floor length	physics curved ramp floor length	Horizontal floor before the arc; default 3.75cm.
curved ramp radius	physics curved ramp radius	Circular radius; default 4cm.
curved ramp angle	physics curved ramp angle	Sweep angle; default 90 degrees.
curved ramp back extension	physics curved ramp back extension	Horizontal body after the arc; default 1cm.
ramp depth	physics ramp depth	Material below the low endpoint; default 0.25cm.
ramp direction	physics ramp direction	right or left.
ramp guide length	physics ramp guide length	Tangent/normal guide length; default 0.5cm.

3.3 Surface, tangent, and normal anchors

Both ramp shapes expose surface-0 through surface-100. The short forms .0 through .100 name exactly the same points. On the straight ramp the parameter follows arc length over the floor and then the incline; it deliberately excludes the vertical wall. On the curved ramp it follows arc length over the new horizontal floor and then the circular arc. Use curve-0 . . 100 when a fraction of the circular portion alone is needed. The straight ramp also has floor, incline, wall-surface, wall-back, wall-tip, base, and end percentage families. The curved ramp has floor, start, base, back, and top families alongside

curve. Together these cover every drawn boundary. The complete visual map is in docs/ramp-anchors.md and docs/ramp-anchor-coverage.pdf.

At every integer T , three orientation helpers are available:

Anchor	Meaning
tangent-before-T	Half a guide length before surface-T along the local tangent.
tangent-after-T	Half a guide length after surface-T along the local tangent.
normal-T	One guide length outward from surface-T, perpendicular to the surface.
curve-tangent-before-T	Half a guide length before curve-T.
curve-tangent-after-T	Half a guide length after curve-T.
curve-normal-T	Outward normal at curve-T only.

The tangent pair is symmetric about surface-T. Consequently TikZ can position and rotate a block without manually calculating its angle:

```
\path (R.tangent-before-60) -- (R.tangent-after-60)
  node[midway,sloped,block,anchor=south] (B) {$m$};
\draw[->] (B.north) --
  ($ (B.north)+(R.normal-60)-(R.surface-60)$) node[above left] {$N$};
```

Named anchors provide the structural points:

Shape	Named anchors
Both	center, surface-start, surface-mid, surface-end, base-start, base-end.
Straight ramp	floor-start, floor-mid, floor-end, ramp-foot, ramp-mid, ramp-top, wall-top, wall-mid, wall-bottom, angle-baseline, angle-label.
Curved ramp	floor-start, floor-mid, floor-end, curve-start, curve-mid, curve-end, curve-center, back-top, back-bottom.

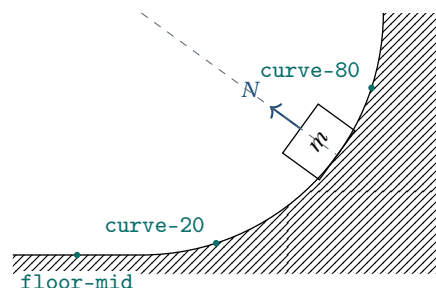
Here is a complete curved-ramp document. The dashed radius is perpendicular to the tangent. The blue force arrow uses the exact outward-normal vector, translated to begin at the block's north anchor so it does not cross the body.

```

\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}
\begin{document}
\begin{tikzpicture}[>=stealth]
  \node[curved-ramp, curved ramp radius=4cm,
    curved ramp angle=90,
    curved ramp back extension=1cm] (R) at (0,0) {};

  \path (R.curve-tangent-before-60) -- (R.curve-tangent-after-60)
    node[midway, sloped, block, minimum width=1cm,
      minimum height=0.75cm, anchor=south] (B) {$m$};
  \draw[->, blue, thick] (B.north) --
    ($ (B.north) + (R.curve-normal-60) - (R.curve-60) $)
    node[above left] {$N$};
  \draw[dashed] (R.curve-center) -- (R.curve-60)
    node[midway, left] {$r$};
\end{tikzpicture}
\end{document}

```



3.4 Styling and debug use

The body follows ordinary TikZ styling. Place the shape style first, then override pattern, pattern color, fill, or draw in the usual option order. The contact line is always emitted once by the shape. For example, `curved-ramp, pattern=grid, pattern color=gray` changes the material rendering without changing anchors.

Debug overlays are especially useful for unfamiliar ramp geometry:

```

\node[curved-ramp, curved ramp floor length=1.5cm,
  curved ramp radius=2.5cm,
  curved ramp angle=70,
  physics debug/anchors=true,
  physics debug/keys=true] (R) {};

```

The key panel presents minimum width before package-specific sizing on the straight ramp. Anchor overlays show a compact representative set rather than all 101 samples. Use explicit `surface-T`, `tangent`, and `normal` anchors when inspecting a particular value.

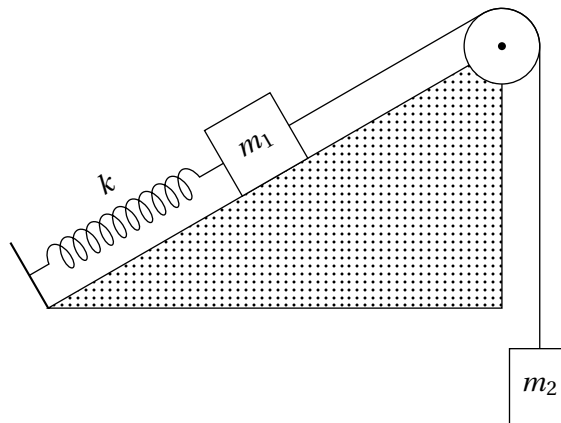
4 Blocks, springs, and pulleys

The `tikzphysics.mechanics` library supplies block and pulley nodes, a spring path style, and tangent pulley strings. They accept ordinary TikZ styling such as `draw`, `line width`, and `scale`.

4.1 Complete spring–block–pulley system

This example combines reusable defaults, a surface tangent, and the native `to[over pulley=P]` connection. Copy the complete document below.

```
\documentclass[tikz,border=6mm]{standalone}
\usepackage{tikzphysics}
\tikzset{
  every block/.style={minimum width=1cm,minimum height=1cm,fill=white},
  every pulley/.style={minimum size=1cm,fill=white},
  every spring/.style={pre length=3mm,post length=3mm,
    amplitude=2mm,segment length=2mm}
}
\begin{document}
\begin{tikzpicture}
  \node[wedge,minimum width=6cm,wedge angle=30] (W) {};
  % The contact anchors select the actual incline in every wedge mode.
  \path (W.tangent-before-50) -- (W.tangent-after-50)
    node[midway,sloped,block,anchor=south] (B) {$m_1$};
  \edef\InclineAngle{\geometryvalue{W}{slope angle}}
  \draw[thick] (W.surface-start) -- ++({\InclineAngle+90}:10mm);
  \coordinate (S) at ($(W.surface-start)+({\InclineAngle+90}:5mm)$);
  \draw[spring] (S) -- node[above=3mm,sloped] {$k$} (B.west);
  \node[pulley] (P) at (W.top) {};
  \node[block,minimum width=8mm,anchor=north] (H) at ($(P.east)+(0,-4cm)$) {$m_2$};
  \draw[rope] (B.east) to[over pulley=P] (H.north);
\end{tikzpicture}
\end{document}
```



The 10 mm block height matches the 10 mm pulley diameter. The hanging block has a local 8 mm width for readability; the package default remains a 1 cm square. With the pulley centred at the top vertex, the incoming rope is parallel to the incline. The support's attachment is half the block height above the surface, so the spring is parallel too. Changing one of these dimensions independently requires adjusting the placement. This example is specifically the rising `br wedge` arrangement; contact anchors support the other modes, but a complete pulley layout must still be chosen for the intended route.

4.2 Native rope connections

```
\draw[rope] (A) to[over pulley=P] (B);
\draw[rope,string route=under,red]
  (A) to[over pulley=P] node[right] {$T$} (B);
```

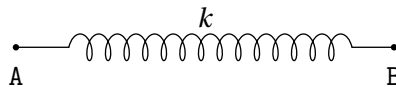
Both endpoints must lie outside the pulley. Use a circular pulley, without nonuniform scaling. The `to` segment computes both tangencies and the contact arc; its labels belong to the final straight segment. Place labels on another segment explicitly when that is what your diagram requires. The `compatibility` command uses the same route:

```
\physicsstringoverpulley[string route=under]{A}{P}{B}
```

Both forms now respect scoped route settings. `shortest` selects the shorter arc for the current tangent pair, not the globally shortest path over every possible tangent combination. The safe path names are `physics string` and `physics over pulley`.

4.3 Spring nodes and paths

`spring` is a drawing style used directly between two coordinates. The path endpoints determine its complete length and direction, so horizontal, vertical, and inclined springs all use the same syntax. Zero-length leads are supported. Straight leads of `pre length` and `post length` surround the coil; the remaining keys map directly to TikZ's standard coil decoration parameters. Use explicit units for typographic sizes such as 5pt; bare convenience-key values are interpreted as centimetres.



For a spring path, use its endpoints and ordinary TikZ nodes placed along it. Alternatively, use the same style in a positioned node:

```
\node[spring,minimum width=3cm,rotate=30] (S) {};
\draw (A) -- (S.start);
\draw (S.end) -- (B.west);
```

The node defaults to a length of 3 cm. Set `minimum width` or `spring length` to change it; `rotate` sets its direction. Both APIs use the same coil decoration and every `spring` hook. Node anchors `start` and `end` lie exactly at the axis attachments; `coil-start` and `coil-end` delimit the straight leads. `axis-0..100` samples the straight axis from start to end, not the coil wire. Compass anchors bound the coil; `minimum height` can reserve more space but does not change its amplitude. Lead lengths must sum to less than the node length. Put labels above or below the node to keep them off the coil.

```
\node[spring,show anchors,show keys,
  physics debug/anchor families={axis},
  physics debug/anchor samples={0,50,100}] (S) {};
```

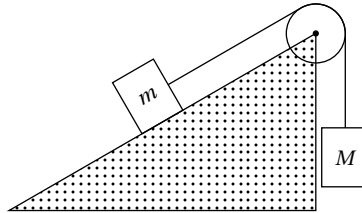
For a wall–spring–block system, a path can connect directly from a chosen wall coordinate to the block anchor:

```

\documentclass[tikz,border=5mm]{standalone}
\usepackage{tikzphysics}
\begin{document}
\begin{tikzpicture}[>=stealth]
  \node[platform-left, minimum width=6cm, minimum height=2cm,
    anchor=surface] (P) at (0,0) {};
  \node[block, minimum width=1cm, minimum height=0.8cm,
    anchor=south] (B) at (P.surface-75) {$m$};
  \coordinate (A) at ($(B.west)+(-4cm,0)$);
  \draw[spring,
    pre length=5pt, post length=5pt,
    amplitude=4.5pt, segment length=4.5pt]
    (A) -- node[above=3pt] {$k$} (B.west);
  \draw[->,thick] (B.east) -- ++(1.1,0) node[right] {$v$};
\end{tikzpicture}
\end{document}

```

4.4 Blocks, pulleys, and tangent strings

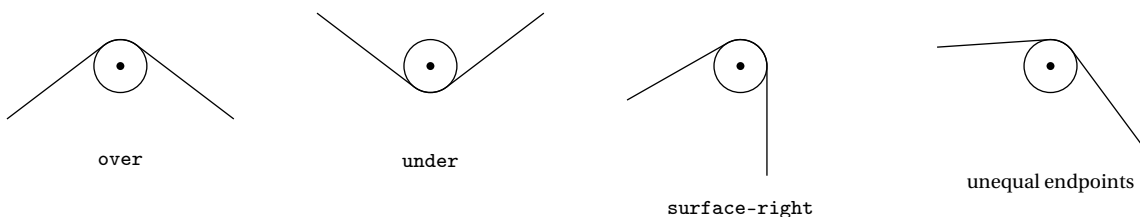


The pulley centre is exactly `W.top`. Its default diameter is 1cm; matching the surface block's height to that diameter places `B.east` one radius above the slope, so the approaching tangent remains exactly parallel to the incline.

`\physicsstringoverpulley{start}{pulley}{end}` draws a single physical string. Both straight portions meet the circular pulley tangentially, and the intervening portion follows the pulley arc. The default surface-right route matches the layout shown above: the pulley is just beyond the incline's right end, the first string portion is parallel to the slope, the contact arc passes over the upper rim, and the second portion descends vertically on the right.

4.5 String-routing variants

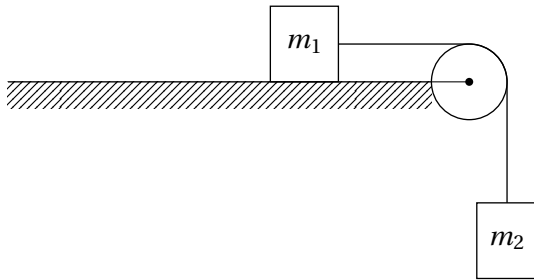
The endpoint coordinates may be placed at any unequal positions outside the pulley. The named routes below include two common left-to-right wraps and the default surface-to-right-hanging turn. `shortest` keeps the selected tangent pair and chooses the shorter contact arc. For other endpoint orderings, select tangent solutions and wrap direction explicitly.



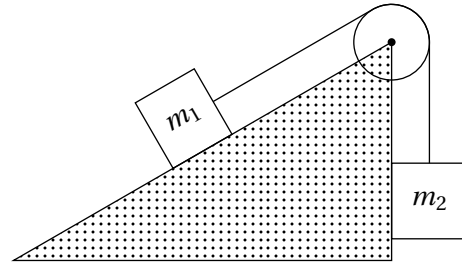
Concise key	Collision-safe alias	Meaning and default
block width	physics block width	Convenience alias for minimum width; default 1cm.
block height	physics block height	Convenience alias for minimum height; default 1cm.
pre length / post length	physics spring pre length / physics spring post length	Straight attachment leads; each defaults to 5pt.
amplitude	physics spring amplitude	Coil half-height; default 4.5pt.
segment length	physics spring segment length	Axial length of one repeated coil segment; default 4.5pt.
aspect	physics spring aspect	Coil projection factor; default 0.5.
pulley diameter	physics pulley diameter	Convenience alias for minimum size; default 1cm.
pulley axle radius	physics pulley axle radius	Axle-dot radius; default 1.5pt. Use none to hide the dot.
pulley axle color	physics pulley axle color	Axle-dot colour; default black.
string start solution	physics string start solution	Start-side tangent, 1 or 2; default 2.
string end solution	physics string end solution	End-side tangent, 1 or 2; default 1.
string route	physics string route	surface-right (default), over, under, or shortest.
string wrap	physics string wrap	clockwise (default), counterclockwise, or shortest.

4.6 Complete pulley systems

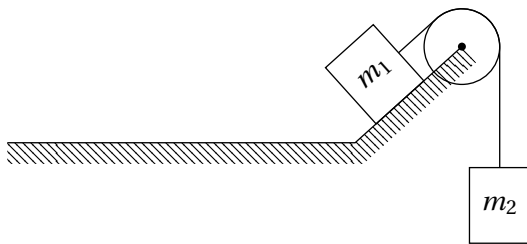
The mechanics shapes compose directly with every surface family. In each diagram below, one call to `\physicsstringoverpulley` draws both straight tangent portions and the circular contact arc; no pulley compass anchor is used as a substitute for the contact point. All four diagrams omit a pulley sizing key and therefore use the documented default `minimum size=1cm`. The inclined and bent systems place the pulley centre directly at the endpoint anchors `W.top` and `S.wall-surface-100`. Their block `height=1cm` makes `B.east` exactly one pulley radius from the supporting surface, so the incoming tangent is surface-parallel.



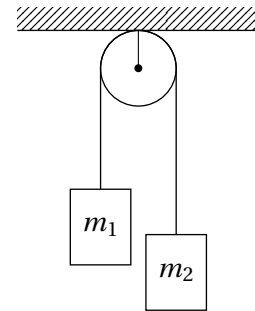
Horizontal plane



Inclined plane



Continuous bent platform



Two hanging masses

Copy-ready standalone sources are included in the examples directory:

- `mechanics-pulley-flat-plane.tex` and `mechanics-block-pulley.tex`;
- `mechanics-pulley-bent-platform.tex` and `mechanics-pulley-two-masses.tex`;
- `mechanics-spring-block.tex`, a complete wall-spring-block system;
- `mechanics-pulley-string-variants.tex`, a gallery of six endpoint and wrapping arrangements.

4.7 Bodies, forces and named assemblies

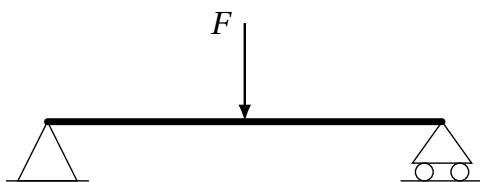
Feature	Use and defaults
particle	Filled circular node, diameter 3 mm.
disk, ring	Circular nodes, diameter 15 mm. Disk has a white fill; ring is unfilled with a thicker outline.
force, acceleration	velocity, Arrow path styles; you choose origin, direction, length, and labels.
torque	Arrow style for an arc you specify.
rod	Straight or curved path with a 2 pt stroke and round caps.
pin-support, roller-support	Named pics, controlled by support size (6 mm). The pic origin is the pivot.
pendulum	Named pic with a particle bob. Length defaults to 2 cm, angle to 30 degrees from downward vertical, positive towards the right.

Circular bodies have ordinary compass and angular anchors. Use `D.center` for an axle and `D.south` for bottom contact in an unrotated disk. These are two-dimensional diagram symbols, not automatic mass-distribution or rigid-body solvers.

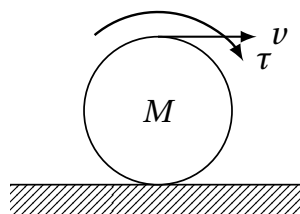
```
\pic (A) {pin-support};
\pic (B) at (4,0) {roller-support};
\draw[rod] (A-pivot) -- (B-pivot);
\draw[force] (2,1) -- (2,0) node[pos=0,left] {$F$};
\pic[pendulum length=2cm,pendulum angle=25] (P) at (6,1) {pendulum};
\draw[force] (P-bob.south) -- ++(0,-1) node[below] {$mg$};
```

Support coordinates are `A-pivot`, `A-base`, `A-left`, and `A-right`; base is the ground line's midpoint, while left/right are the triangle's bottom corners. A pendulum exposes `P-pivot` and a real bob node `P-bob`. Its specified length is pivot to bob centre; the rope stops at the bob's border. Rotate or shift a pic using normal TikZ options. Pic coordinates use a hyphen after the pic name, unlike a node's `B.west` anchor syntax.

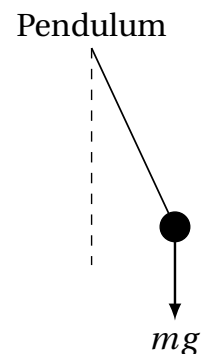
every support and every pendulum are applied when the pic type is expanded. As with TikZ pics generally, type styles are processed after options preceding the type; avoid conflicting geometric values in these hooks and the pic options. Use the hooks for shared appearance or set support size and pendulum length in the surrounding scope. The collision-safe pic types are `physics pin-support`, `physics roller-support`, and `physics pendulum`. See `examples/mechanics-primitives.tex` for a complete gallery.



Supports and a rod



Disk, velocity and torque

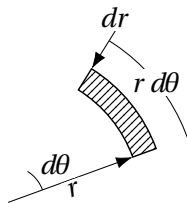


Pendulum

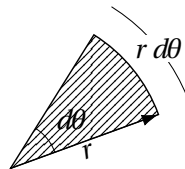
5 Circular differential elements

Element labels inherit the surrounding document or native TikZ font, including per-node font overrides. Dimensions use font-relative arrow tips and spacing; every element label and every element dimension remain user-overridable. `examples/elements-fonts.tex` illustrates this.

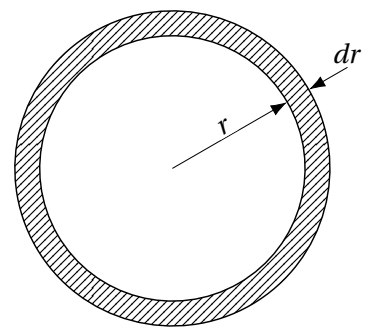
The `tikzphysics.elements` library turns the shaded regions used in polar-coordinate integration and moment-of-inertia derivations into reusable TikZ nodes. The general `polar element` is an annular sector. Its zero-radius and full-ring limits are available as named presets.



polar element



differential sector



differential ring

5.1 General polar element

Use explicit inner and outer radii, or give an inner radius and radial thickness. The later of `element outer radius` and `element radial thickness` controls the outer boundary.

```
\begin{tikzpicture}
  \node[polar element,
    element inner radius=18mm,
    element radial thickness=2mm,
    element start angle=30,
    element delta angle=30,
    show dimensions] (dA) at (0,0) {};
\end{tikzpicture}
```

The inner radius may be zero. The outer radius must be larger than the inner radius. The angular extent must lie in the interval $(0, 360]$ degrees. Bare radial values are centimetres; explicit TeX dimensions retain their unit.

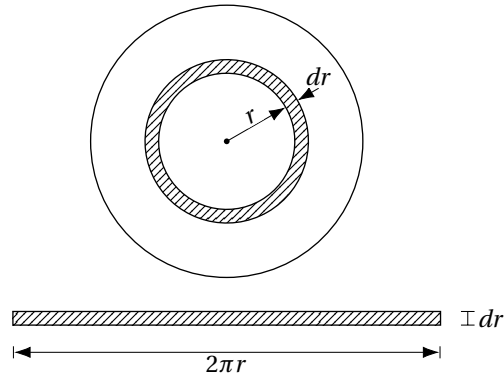
5.2 Sector, ring, and unwrapped strip

`differential sector` fixes the inner radius at zero. `differential ring` fixes the angular extent at 360 degrees and produces a closed annulus without a radial seam. The differential strip uses the inner reference radius, so its width is $2\pi r$ and its height is dr .

```
\node[differential ring,
  element inner radius=18mm,
  element radial thickness=2mm] (R) {};
\node[unwrapped ring,source element=R,
  show dimensions] (U) at (0,-4) {};
```

The named pic keeps the original circular body, annular element, centre mark, and opened strip in one construction. It exposes both component nodes:

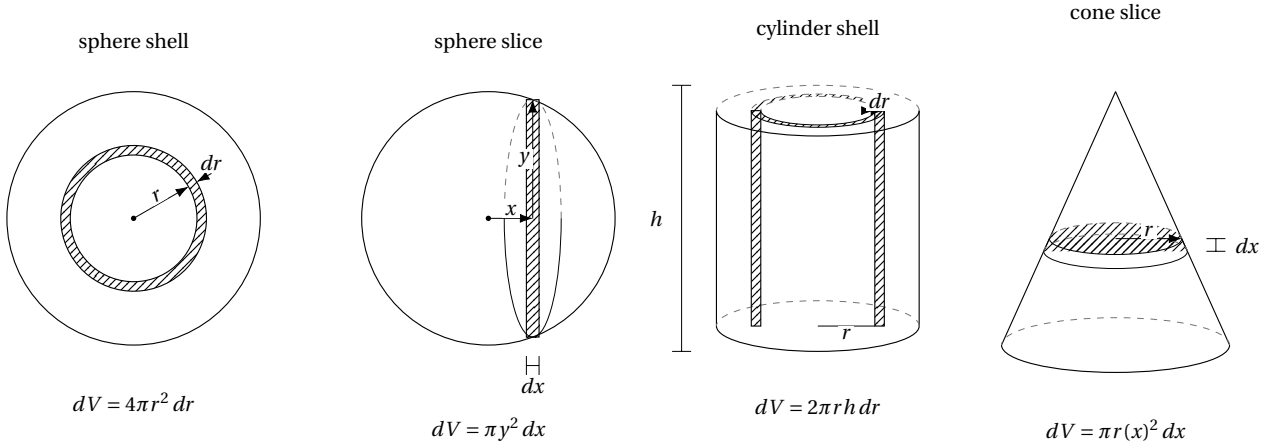
```
\pic (D) {differential ring diagram={
  element inner radius=1cm,
  element radial thickness=2mm
}};
% Refer to (D-ring), (D-strip), and (D-body-east), etc.
```



The body radius defaults to 2 cm, the centre-dot radius to 1.2 pt, and the gap above the strip to 5 mm. Set them with `element body radius`, `element center dot radius`, and `element diagram gap`. The body coordinates are `D-center`, `D-body-north`, `D-body-south`, `D-body-east`, and `D-body-west`.

5.3 Solid, shell, slice, and sheet constructions

The integration method is part of each public name. A solid sphere can use a concentric shell or a circular disk; a cylinder can use a radial shell or an axial disk. Each construction is a named pic containing separately named body and element components.



Construction	Public pic	Default expression
Solid sphere, shells	sphere shell diagram	$dV = 4\pi r^2 dr$
Solid sphere, disks	sphere slice diagram	$dV = \pi y^2 dx$
Finite hollow sphere	hollow sphere diagram	$V = \frac{4\pi}{3}(R^3 - a^3)$
Cylinder, shells	cylinder shell diagram	$dV = 2\pi r h dr$
Cylinder, disks	cylinder slice diagram	$dV = \pi R^2 dz$
Cone, disks	cone slice diagram	$dV = \pi r(x)^2 dx$
Cartesian sheet	sheet element diagram	$dA = dx dy$

```

\begin{tikzpicture}
  \pic (S) {sphere shell diagram={
    element body radius=2cm,
    element inner radius=1cm,
    element radial thickness=1.5mm
  }};
  \pic (C) at (6,0) {cylinder slice diagram={
    element body radius=1.6cm,
    element body height=3.4cm,
    element position=.55,
    element axial thickness=2mm
  }};
\end{tikzpicture}

```

On sphere slice diagram, element position is the signed axial coordinate divided by the sphere radius and lies strictly between -1 and 1 . On cylinder diagrams the same key runs from bottom to top in $[0,1]$. On cone slice diagram it runs from apex to base in $(0,1)$. element projection ratio controls the apparent depth of the projected circles without changing their physical radii.

For a pic named S, use S-body and S-shell or S-slice. Every pic also provides S-center and S-formula-anchor; cone pics add S-apex and S-base. The sheet construction uses S-body and S-element. Override the displayed expression with element formula label.

The reusable nodes are spherical shell, hollow sphere, rectangular element, rectangular strip, and rectangular sheet:

```
\node[spherical shell,show dimensions] (S) {};
\node[hollow sphere,element inner radius=8mm,
      element outer radius=2cm] (H) at (5,0) {};
\node[rectangular element,element width=8mm,
      element height=6mm,show dimensions] (dA) at (10,0) {};
```

The spherical nodes expose the polar boundary anchors and numeric families. Rectangular nodes expose their four corners and the families `bottom-0..100`, `right-0..100`, `top-0..100`, and `left-0..100`. Named rectangles record resolved width and height. Use `show anchors`, `show keys`, or a card such as `\physicshelp{cylinder slice diagram}` inside the picture. The labels use plain LaTeX dx , dy , and dr .

Dimension labels and arrow sizes inherit the surrounding or per-node font. On an element node, users can write `pattern=dots`, `pattern=north east lines`, `pattern=horizontal lines`, `dashed`, or `pattern=none`, `fill=gray!20`. For the highlighted slice or shell inside a solid pic, put the same native keys in every solid element/.append style={...}. Projected circular bodies draw rear curves through every solid hidden edge; the visible front curves use the ordinary body or dimension style. See `examples/elements-fonts.tex` for rendered font and pattern variants.

5.4 Labels, anchors, and resolved values

`show dimensions` uses plain LaTeX labels: r , dr , $d\theta$, $r d\theta$, and $2\pi r$. Set the radial labels with `element radius label` and `element radial label`. Set the angular, arc, and strip labels with `element angular label`, `element arc label`, and `element circumference label`. Shared appearance hooks are `every element dimension` and `every element label`; the combined pic also provides `every element body`, `every element center`, and `every differential ring diagram`.

The principal anchors are `inner-start`, `inner-mid`, `inner-end`, `outer-start`, `outer-mid`, `outer-end`, `start-mid`, `end-mid`, `center`, and the true area centroid. Four numeric families—`inner-0..100`, `outer-0..100`, `start-0..100`, and `end-0..100`—sample every boundary. A complete ring's centroid coincides with its centre.

Named polar elements record `inner radius`, `outer radius`, `radial thickness`, `mean radius`, `start angle`, `delta angle`, and `end angle`. An unwrapped ring also records `circumference`. Query them with `\geometryvalue`:

```
\node at (R.center)
  {$r=\geometryvalue{R}{mean radius}$};
```

Use `show anchors`, `show keys`, or `\physicshelp{polar element}` to inspect the API inside the picture. See `examples/elements-circular-differentials.tex` for the complete standalone gallery.

6 Fluid mechanics

The separate `tikzphysics.fluids` library provides eight native nodes and fourteen editable teaching assemblies. These are geometric schematics: levels, curvature, and dimensions are inputs, not computed fluid equilibria. Load the complete package with `\usepackage{tikzphysics}`. To load only this module:

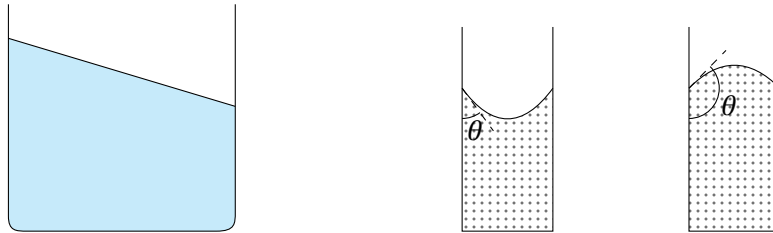
```
\usepackage{tikz}
\usetikzlibrary{tikzphysics.fluids}
```

```
\begin{tikzpicture}[font=\small]
  \node[fluid tank,
```

```

fluid={width=4cm,height=3cm,left level=.85,right level=.55},
pattern=north east lines,pattern color=gray,line width=.7pt
] (T) {};
\draw[->] (T.right-surface) -- ++(1,0) node[right] {$a_x$};
\end{tikzpicture}

```



6.1 Native nodes and patterns

Use `\node[fluid tank] (T) {};` for an individual object. The native node styles are fluid tank, fluid cylinder, pressure element, meniscus, rotating fluid, flow tube, liquid ring, and u tube. They support standard compass, base, and mid anchors, text, anchor, minimum width, minimum height, inner sep, outer sep, and the positioning library. Native minimum sizes enlarge the frame; fluid dimensions set its base size. Nodes are centred on their placement coordinate.

Use semantic node anchors such as `(T.right-surface)`, `(P.inlet)`, and `(R.liquid-start)` to attach ordinary paths. Automatic connections use the nominal frame, circular for liquid rings. Cylinder-node levels are measured between end-cap centres. U-tube nodes show one liquid; the assembly adds the second liquid and interface labels.

Default liquid shading is a light dot pattern. It can be changed directly:

```

\node[fluid tank,pattern=north east lines,pattern color=gray] {};
\node[fluid tank,pattern=horizontal lines,dashed] at (4,0) {};
\node[fluid tank,fill=cyan!20] at (8,0) {}; % solid liquid
\node[fluid tank,pattern=none] at (12,0) {}; % outline only

```

A later fill replaces a pattern; a later pattern selects a pattern. Dashes affect outlines and may affect stroke-based patterns as in native PGF. The fill and pattern handlers remain local to each fluid node. See `examples/fluid-patterns.tex` for simultaneous patterns and ordinary TikZ nodes. `examples/fluid-mechanics-nodes.tex` shows all eight nodes.

6.2 Composition and native styling

For a complete assembly, put geometry inside the explicit pic argument:

```

\pic (T) {fluid tank diagram={fluid width=4cm}};

```

Use ordinary `at`, `rotate`, `scale`, and `transform shape` for placement and transformations. Native `draw`, `fill`, `color`, `line width`, `opacity`, and `font` are supported. Text and mathematics inherit the document fonts; default arrows and label padding follow font size. Geometry remains in physical lengths.

The scoped spelling `fluid={width=4cm}` and the collision-safe spelling `physics fluid width=4cm` mean the same thing. Every `pic` also has an explicit physics alias, such as `physics fluid tank diagram`. Native global keys are not redefined. Unitless lengths mean centimetres; levels are fractions of height. A repeated preset can use `every fluid tank/.append style={...}`.

Every `pic` named `F` exposes coordinates `F-origin`, `F-center`, `F-bottom`, `F-top`, `F-left`, and `F-right`. These describe the nominal frame. Semantic coordinates such as `F-left-surface`, `F-large-piston`,

F-contact-left, and F-inlet appear in the feature reference. Use (F-surface), not node-anchor syntax. `\physicshelp{fluid tank diagram}` displays assembly defaults and coordinates. The shorter `fluid tank help` entry describes the node. The `show anchors` overlay works on fluid nodes; it does not apply to pics. Where both interfaces exist, the explicit pic name ends in `diagram`, for example `fluid tank diagram`.

6.3 Diagram families

- Hydrostatics: `u tube`, `gas manometer`, `hydraulic press`, `fluid cylinder`, and `fluid tank`. Unequal tank endpoint levels represent an accelerating liquid; equal levels give a horizontal surface.
- Pressure and rotation: `pressure element`, `rotating tube`, `rotating fluid`, and `liquid ring`. The rotating-vessel surface is parabolic; its `fluid bend` is the wall-to-vertex rise divided by height.
- Flow and buoyancy: `flow tube` and `buoyancy`. The outlet is 0.55 times the inlet height. Pressure-arrow lengths are schematic.
- Interfaces: `meniscus`, `capillary`, and `surface tension ring`. Positive meniscus bend gives a concave surface; negative bend gives a convex surface. The contact-angle arc is measured through the liquid.

6.4 Labels, hooks, and geometry limits

Generic labels are `fluid body label`, `fluid left label`, `fluid right label`, and `fluid dimension label`. Named labels include `fluid force label`, `fluid volume label`, `fluid pressure label`, `fluid pressure increment label`, `fluid angular velocity label`, `fluid position label`, `fluid horizontal axis label`, `fluid vertical axis label`, `fluid inlet velocity label`, `fluid outlet velocity label`, `fluid contact angle label`, `fluid rise label`, `fluid tension label`, `fluid circumference force label`, and `fluid radius label`. Empty text suppresses a label. All accept arbitrary LaTeX.

Style hooks are `every fluid outline`, `every fluid liquid`, `every fluid secondary liquid`, `every fluid label`, `every fluid arrow`, and `every fluid dimension`; each also has an `every physics fluid` alias. For example:

```
\tikzset{every fluid arrow/.append style={>={Stealth[length=.7em]}},
every fluid label/.append style={font=\footnotesize}}
```

U-tube levels must be above the bend, and tube width below one third of both frame dimensions. Hydraulic piston heights must exceed tube width, which must also be below 0.3 times frame width. The rotating-tube section must fit inside its length. The rotating vessel requires nonnegative bend and `level+bend` at most one. Meniscus level plus or minus absolute bend must stay in $[0, 1]$. Capillary rise requires nonnegative bend and `left level - 0.75*bend` at least the reservoir level. The capillary height annotation measures to the contact line. Ring thickness must be below half the diameter, with sweep in $(0, 360]$. Invalid geometry raises a package error.

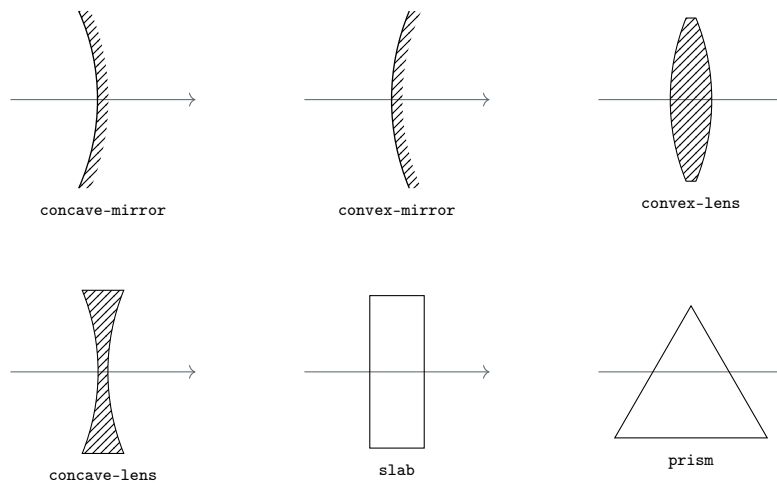
The complete usage guide is `docs/fluids.md`. Copy-ready sources:

- `examples/fluid-mechanics-gallery.tex`
- `examples/fluid-mechanics-reference-scenes.tex`

The latter covers all thirteen reference compositions. The generated single-file Overleaf package includes the module automatically.

7 Optical components

The `tikzphysics.optics` library provides six node styles. They remain ordinary TikZ nodes, so rotation, scaling, styling, naming, and anchor-based composition work in the usual way.



Short style	Collision-safe style	Geometry
concave-mirror	physicsconcavemirror	Concave reflecting surface
convex-mirror	physicsconvexmirror	Convex reflecting surface
convex-lens	physicsconvxlens	Symmetric biconvex element
concave-lens	physicsconcavelens	Symmetric biconcave element
slab	physicsslab	Parallel rectangular faces
prism	physicsprism	Isosceles triangular cross-section

7.1 Concave and convex mirrors

concave-mirror and convex-mirror are reflected variants of one spherical-mirror cross-section. The hatched side represents the backing; the solid arc is the reflecting surface.

Key	Meaning and default
mirror radius	Radius of the reflecting surface; default 5cm.
mirror thickness	On-axis backing thickness; default 0.25cm.
mirror aperture angle	Reflecting-surface half-angle; default 30 degrees.

The aperture angle must be strictly between 0 and 90 degrees. Named anchors include surface-mid, surface-top, surface-bottom, back-mid, back-top, back-bottom, vertex, top, and bottom. The families surface-0..100 and back-0..100 run from bottom to top.

Collision-safe aliases add the physics prefix:

- physics mirror radius;
- physics mirror thickness;
- physics mirror aperture angle.

```

\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}
\begin{document}
\begin{tikzpicture}
  \node[concave-mirror, mirror radius=5cm,
        mirror aperture angle=28] (M1) at (0,0) {};
  \node[convex-mirror, mirror radius=5cm,
        mirror aperture angle=28] (M2) at (5,0) {};
  \draw[>-] (-2,0.7) -- (M1.surface-70);
  \draw[>-] (3,0.7) -- (M2.surface-70);
\end{tikzpicture}
\end{document}

```

7.2 Convex and concave lenses

`convex-lens` draws a symmetric biconvex element, while `concave-lens` draws a symmetric biconcave element.

Key	Meaning and default
<code>convex lens radius</code>	Radius of both optical surfaces; default 5cm.
<code>convex lens thickness</code>	Material thickness at the top and bottom caps; default 0.2cm.
<code>convex lens aperture angle</code>	Half-angle of both surfaces; default 25 degrees.
<code>concave lens radius</code>	Radius of both optical surfaces; default 5cm.
<code>concave lens thickness</code>	Material thickness on the optical axis; default 0.2cm.
<code>concave lens aperture angle</code>	Half-angle of both surfaces; default 25 degrees.

The aperture angle must be strictly between 0 and 90 degrees. Named anchors include `front-mid`, `back-mid`, `front-top`, `front-bottom`, `back-top`, `back-bottom`, `top`, and `bottom`. The arc families `front-0..100` and `back-0..100` run from bottom to top. Collision-safe keys add the `physics` prefix, for example `physics convex lens radius` and `physics concave lens radius`.

7.3 Slab

`slab` uses native `minimum width` and `minimum height`; the unit-aware conveniences `slab width` and `slab height` set the same requirements. Defaults are 1.2cm by 3cm. The shape is an unfilled outline, so ray paths remain visible through it.

Named anchors are `front`, `back`, `front-top`, `front-bottom`, `back-top`, `back-bottom`, `top`, and `bottom`. Numeric families `front-0..100` and `back-0..100` run bottom to top; `top-0..100` and `bottom-0..100` cover the caps.

7.4 Prism

`prism` is an isosceles triangle. Use native `minimum width` and `minimum height`, or the unit-aware `prism width` and `prism height`. Defaults are 3cm by 2.6cm. If `prism apex angle` is supplied, it is authoritative and the height is derived from the width; the angle must be strictly between 0 and 180 degrees. The prism is also an unfilled outline.

Named anchors are `apex`, `base-left`, `base-right`, `base-mid`, `left-mid`, `right-mid`, `entry-mid`, `exit-mid`, and `centroid`. Numeric families are `base-0..100`, `left-0..100`, and `right-0..100`.

7.5 Short numeric anchors

Each optical shape also exposes its primary ray-entry surface directly as `.0` through `.100`. Thus `(L.50)` is the midpoint and `(L.80)` is a point near the top of the primary surface.

Shape	Meaning of <code>(name.T)</code> for $T = 0 \dots 100$
<code>concave-mirror /</code> <code>convex-mirror</code>	Same as <code>surface-T</code>
<code>convex-lens /</code> <code>concave-lens</code>	Same as <code>front-T</code>
<code>slab</code>	Same as <code>front-T</code>
<code>prism</code>	Same as <code>left-T</code>

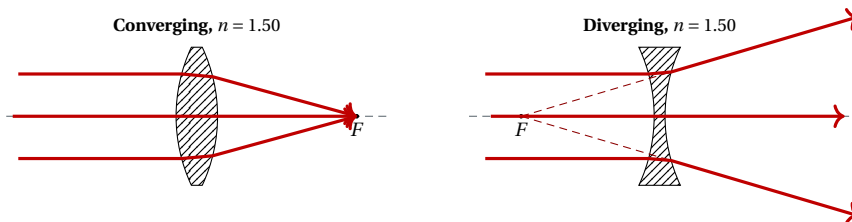
Explicit family names can mark corresponding points, for example `(L.80) - (L.back-80)`; that segment is not generally a physical ray.

7.6 Complete copy-paste ray diagram

For the fixed ratio $n_{\text{lens}}/n_{\text{air}} = 1.50$, the following Snell-law construction bends each non-axial ray at both interfaces. The axial ray stays straight because it meets both vertices normally.

```
\documentclass[tikz, border=6mm]{standalone}
\usepackage{tikzphysics}

\begin{document}
\begin{tikzpicture}[ray/.style={red!75!black, very thick, ->}]
  \node[convex-lens, convex lens radius=3cm,
    convex lens thickness=0.2cm,
    convex lens aperture angle=25] (L) at (0,0) {};
  \coordinate (Xlow) at ($(L.center)+(0.29194,-0.72588)$);
  \coordinate (Xhigh) at ($(L.center)+(0.29194,0.72588)$);
  \coordinate (F) at (2.93841,0);
  \draw[densely dashed, gray] (-3.5,0) -- (3.5,0);
  \fill (F) circle (1.2pt) node[below] {$F$};
  \draw[ray] ($(L.20)+(-3,0)$) -- (L.20) -- (Xlow) -- (F);
  \draw[ray] ($(L.50)+(-3,0)$) -- (L.50) -- (L.back-50) -- (F);
  \draw[ray] ($(L.80)+(-3,0)$) -- (L.80) -- (Xhigh) -- (F);
\end{tikzpicture}
\end{document}
```



Note. The package supplies geometry and anchors, not automatic ray tracing. These coordinates are valid only for the stated lens and index; recalculate them after changing either. The complete four-panel source is `examples/optics-complete-ray-diagrams.tex`.

8 Shared tools

8.1 Percentage anchors: a beginner's guide

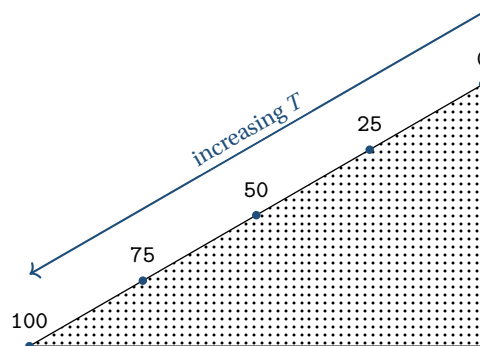
A percentage anchor names a point partway along one particular edge, arc, or contact surface. It removes the need to calculate coordinates manually. The general spelling is

(node-name.family-number)

For example, (W.slope-25) means: use node W, choose its slope edge, then travel 25 percent from that family's documented start towards its end. The number is an integer from 0 to 100; write 25, not 25%. The three landmarks worth memorising are:

family-0	start of the family
family-50	halfway along the family
family-100	end of the family

The family direction matters. On a wedge, slope-0 is the top vertex and slope-100 is the lower foot, so slope-25 lies near the top rather than near the foot.



Using a percentage anchor

The node must first have a name. Then use the anchor anywhere TikZ accepts a coordinate:

```
\begin{tikzpicture}
  \node[wedge, wedge width=6cm, wedge angle=30] (W) at (0,0) {};

  % Put the block midpoint on the middle of the slope.
  % anchor=south makes the block's bottom touch that point.
  \node[block, minimum width=1cm, minimum height=0.7cm,
    rotate=30, anchor=south]
    (B) at (W.slope-50) {$m$};

  % Mark a point 25 percent of the way from the top to the foot.
  \fill[red] (W.slope-25) circle (1.5pt);

  % Draw between two percentage anchors.
  \draw[blue, very thick] (W.base-20) -- (W.slope-80);
\end{tikzpicture}
```

Changing the wedge width or angle moves all three points automatically. The number describes progress along the selected family, not an x coordinate, y coordinate, distance in centimetres, or angle around the node.

Which way does the percentage increase?

Use 0 as the start and 100 as the arrowhead in the table below. Explicit family names are recommended in teaching material because they state both the surface and its direction.

Family	Direction from 0 to 100
surface-T on a platform or ground	left to right across the top contact face
surface-T on a ceiling	left to right across the underside
surface-T on a wall-left or wall-right	bottom to top on the contact face
bottom-T on a platform	left to right across the underside
left-T, right-T on a rectangular surface	bottom to top
wall-surface-T, wall-back-T on a platform	wall root towards its free tip
base-T on a wedge	bl to br
right-T on a wedge	br to top
slope-T on a wedge	top to bl
surface-T on a ramp	surface-start to surface-end, following the complete contact path
curve-T on a curved ramp	start to end of the circular portion only
surface-T, front-T, back-T in optics	bottom to top
base-T on a prism	base-left to base-right
left-T on a prism	apex to base-left
right-T on a prism	base-right to apex

Fluid free-surface families run from left to right. On a U tube, left-surface-T and right-surface-T independently cover the liquid in each arm; surface-T selects the left arm. Block boundaries follow bottom left-to-right, right bottom-to-top, top right-to-left, and left top-to-bottom. Mechanics rim-T starts at the rightmost point and runs counterclockwise: 25 is north, 50 west, and 75 south. Projected cylinders provide a left-to-right surface-T diameter and an elliptical surface-rim-T. Flow tubes expose vertical inlet-T and outlet-T faces and a curved axis-T. Cubic curves use their curve parameter; each family definition appears in the reference below.

For a multi-part ramp, the percentage follows distance along the complete contact path. Consequently surface-50 is halfway by path length; it need not be the floor-to-incline joint or the visual centre of the node. On a circular family the percentage follows the arc.

Short form (name.T)

Ramps and optical shapes also provide a short form such as (R.50) or (L.80). It selects one documented primary surface:

Shape	Meaning of (name.T)
straight or curved ramp	surface-T
concave or convex mirror	surface-T
convex or concave lens	front-T
slab	front-T
prism	left-T

Thus (L.80) and (L.front-80) are the same point. Platforms and wedges intentionally require the explicit family name, such as (P.surface-50) or (W.slope-50). This also avoids confusion with ordinary

TikZ numeric border anchors, where a number can mean an angle around a node rather than a percentage.

The complete annotated source is `examples/percentage-anchors-beginner.tex`; the more exhaustive edge gallery is `examples/surface-numeric-anchors.tex`.

8.2 Combining x and y coordinates

TikZ's native projection operators solve a common mechanics-diagram problem: take the horizontal position from one anchor and the vertical position from another. They avoid measured offsets and continue working when either object moves or changes size.

```
(A |- B)  x-coordinate from A, y-coordinate from B
(A -| B)  x-coordinate from B, y-coordinate from A
```

Suppose block B must connect horizontally to the vertical wall of ramp R. Use the wall anchor only as the x source and B.west as the y source:

```
\node[ramp, minimum width=8.6cm, ramp run=2.6cm,
      ramp rise=1.5cm] (R) at (0,0) {};
\node[block, minimum width=1cm, minimum height=0.75cm,
      anchor=south] (B) at (R.floor-mid) {$m$};

\coordinate (A) at (R.wall-mid |- B.west);
\draw[spring] (A) -- (B.west);
```

Both endpoints now have exactly the same y-coordinate, so the spring or line is horizontal by construction. The same recipe works with an upward platform wall:

```
\node[platform-left-up, minimum width=6cm] (P) at (0,0) {};
\node[block, anchor=south] (B) at (P.surface-75) {$m$};
\coordinate (A) at (P.wall-surface-50 |- B.west);
\draw[spring] (A) -- (B.west);
```

For a right upward wall, use B.east and a right-wall anchor instead. This projection is exact for vertical walls. For an angled wall it keeps the chosen anchor's x-coordinate; finding the geometric intersection with an arbitrary sloping wall instead requires TikZ's intersections library.

See examples/coordinate-projection.tex for both complete diagrams.

8.3 Explore keys and anchors inside TikZ

Add `show anchors` or `show keys` to a node, named or unnamed. Both keys accept `true/false`; a bare key means `true`. The original `physics debug/anchors`, `physics debug/keys`, and `physics debug/all` forms remain supported. The overlays work on every native physics node, including the eight fluid nodes. Paths and pics use `\physicshelp{name}` cards.

```
\begin{tikzpicture}
  \node[wedge, show anchors, show keys] (W) {};
\end{tikzpicture}
```

`show keys` is a complete feature reference, labelled as *defaults*, not a dump of live node values. It lists geometry keys, size aliases, all named anchors, percentage families, and common TikZ styling options. Arbitrary user-defined TikZ keys cannot be enumerated by the package. The full PGF/TikZ manual remains the reference for generic keys.

Object type	In-picture discovery
Node, such as wedge or fluid tank	Add <code>show anchors,show keys</code> to its node options.
Path, such as spring	Use <code>\physicshelp{spring}</code> .
Pic, such as fluid tank diagram	Use <code>\physicshelp{fluid tank diagram}</code> ; its (F-surface) coordinate is not a node anchor.

Use `\physicshelp{index}` to list every feature. Display a reference without constructing a shape:

```
\begin{tikzpicture}
  \physicshelp{spring}
\end{tikzpicture}
```

`\physicshelp[<node options>]{<feature>}` accepts every public feature name, including paths and pics. You can place a card with `xshift`, `yshift`, `anchor`, or `normal` at node options. Use `\tikzphysicshelp` for the collision-safe command name. The reference cards also work when loading only one feature module. A card can describe another module's feature, but you must load that module before drawing it. For complete rendered examples, compile `examples/debug-feature-tour.tex`. Its pages cover a platform, wedge, curved ramp, fluid tank, mirror, Cartesian element, path, and pic. The companion `docs/debug-overlays.md` explains all debug options and links the detailed surface, wedge, ramp, and fluid/mechanics anchor galleries. The latter is `examples/fluid-mechanics-anchor-coverage.tex`.

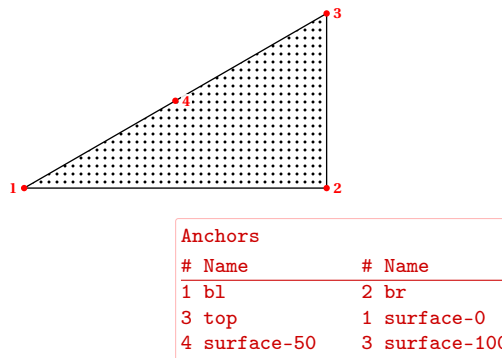
```
% Inspect a ramp's floor, arc, and back wall.
\node[curved-ramp,show anchors,show keys,
  physics debug/anchor list={surface-start,curve-start,curve-end},
  physics debug/anchor families={floor,curve,back},
  physics debug/anchor samples={0,50,100}] (R) {};

% A native fluid node uses the same overlays.
\node[fluid tank,show anchors,show keys,
  physics debug/anchor list={surface,left-surface,right-surface},
  physics debug/anchor families={surface,bottom,left,right},
  physics debug/anchor samples={0,50,100}]
(F) at (10,0) {};
```

Choose points without covering the diagram

```
\node[wedge,show anchors,
  physics debug/anchor list={bl,br,top},
  physics debug/anchor families={surface},
  physics debug/anchor samples={0,25,50,75,100}] (W) {};
```

`anchor list=auto` is the default and enumerates all named anchors in a table. One numbered marker is placed at each distinct position; multiple anchors at that position share the same number in the table. An explicit comma-separated list replaces that selection. Families add the chosen samples; for example `surface` plus 50 adds `surface-50`. You may leave the named list empty to inspect only families. To inspect all points of one family, use `anchor samples={0,1,...,100}`. You can select every family with `anchor families=all`; most diagrams are clearer with a few samples. The reference always lists the full family ranges. Family names and samples must exist on that object; unknown anchors receive the normal PGF error. Coincident points share one label with multiple reference numbers.



Scope settings and move panels

```
\begin{tikzpicture}[show anchors]
  \node[block] (A) {};
  \node[block,show anchors=false] (B) at (4,0) {};
\end{tikzpicture}
```

Options are captured for each object, so local false overrides, fonts, and colours remain associated with that object. Debug drawings do not recursively acquire debug overlays. Explicit unique node names are needed only when later commands refer to the object's anchors.

Key under physics debug/	Default and effect
anchors, keys, all	false; bare form enables the overlay.
anchor list	auto; all named anchors, or a comma-separated selection.
anchor families	Empty; selected percentage families, or all.
anchor samples	0,25,50,75,100.
anchor dot color, anchor dot radius	red, 1.2pt.
anchor label color, anchor label font	red!80!black, scriptsize typewriter.
key label color, key label font	gray, tiny typewriter.
anchor legend xshift, anchor legend yshift	0pt from the object's centre, -4mm below the lowest displayed anchor.
key panel xshift, key panel yshift	0pt, 4mm above the highest named anchor or displayed marker.

Panels deliberately enlarge the picture's bounding box. For a complex system, inspect one object at a time or move its reference card beside the system. Disable overlays for the final diagram. The examples `debug-explorer.tex` and `debug-scoped.tex` are complete documents.

8.4 Troubleshooting and geometry limits

My default size did not apply. Put the style first and the size after it, or set every block. Built-in style values follow ordinary option order.

The block is on a vertical wedge edge. Use `surface-T` for physical contact. The older `slope-T` name follows a fixed CCW edge and is not the incline in every wedge mode.

My block is not rotated with its surface. Use the tangent-before/after path with `sloped`. Local scalar geometry values do not include an independent node transformation.

An automatic line ends outside the shape. Irregular shapes still inherit rectangular automatic border behaviour. Use explicit contact anchors; true outline intersections for all irregular bodies are future work.

Scaling makes a normal or rope wrong. Nonuniform scaling changes circles to ellipses and does not preserve perpendicularity. Keep physical constructions rigid or uniformly scaled. Use `transform shape` when scaling the entire node geometry with a scope.

Long text does not fit an irregular shape. Package geometry dimensions define its body; do not assume native text-driven minimum-size growth. Place labels in separate nodes. Blocks and circular primitives use standard TikZ shapes and can grow to fit text.

What does a bare number mean? Convenience lengths use centimetres; native sizing keys should receive explicit units. Angles are degrees.

Is a numeric anchor an angle? On a pulley or disk, yes. On ramps and optical shapes, integer .0 through .100 are retained percentage shortcuts. Explicit family names avoid ambiguity.

Does the package solve the physics? No. Forces, constraints, motion, optical exit points, and physically compatible placements are supplied by you.

9 Installation and development

The installed drawing package requires only LaTeX2e and PGF/TikZ; using it in a document does not require shell escape or an external executable. The source manual uses Fourier and minted, so building `tikzphysics.pdf` also requires those packages and shell escape for syntax highlighting.

For manual installation, place `tikzphysics.sty` and every `tikzlibrarytikzphysics*.code.tex` file in a directory searched by TeX. For a one-file Overleaf upload, run the bundle command shown below and upload only `output/overleaf/tikzphysics.sty`. The generated file contains all runtime modules and the in-picture feature catalog.

The source distribution uses l3build:

```
l3build check    # run coordinate and anchor regression tests
l3build doc      # build tikzphysics.pdf (shell escape is configured)
l3build ctan     # create the CTAN and TDS archives
python3 scripts/build_overleaf_bundle.py # one-file Overleaf package
```

Standalone examples are stored in `examples/`. The command-line helper `tikzphysics-help` prints a compact key and anchor reference.

10 Complete feature reference

Each entry lists built-in defaults, anchors, and available percentage families. Use explicit family names in new diagrams. The same declarations generate `docs/reference.md` and the in-picture `\physicshelp` cards. Common TikZ keys are additional to the geometry keys shown here; see the earlier sections for sizing and transformation behaviour.

10.1 platform-left (node)

Key	Default
<code>minimum width</code>	5cm
<code>minimum height</code>	2cm
<code>strip width</code>	0.25cm
<code>wall angle</code>	-90 degrees
<code>wall inset</code>	0cm
<code>wall drop</code>	0cm
<code>pulley edge</code>	off; preset uses 0.5cm inset and drop

Size aliases: platform width = minimum width; platform depth = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, surface, pulley-center, wall-root, floor-center, wall-center, text

Families: bottom-0..100, right-0..100, surface-0..100, left-0..100, transition-0..100, wall-surface-0..100, wall-back-0..100, wall-base-0..100, wall-tip-0..100

surface runs left to right across the floor; bottom, left, and right cover its other edges. wall-surface and wall-back run from root to tip; wall-base and wall-tip span the end caps. transition runs from pulley-center to wall-root. Upward walls require zero inset and drop. Use 50 for any edge midpoint.

10.2 platform-right (node)

Key	Default
minimum width	5cm
minimum height	2cm
strip width	0.25cm
wall angle	-90 degrees
wall inset	0cm
wall drop	0cm
pulley edge	off; preset uses 0.5cm inset and drop

Size aliases: platform width = minimum width; platform depth = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, surface, pulley-center, wall-root, floor-center, wall-center, text

Families: bottom-0..100, right-0..100, surface-0..100, left-0..100, transition-0..100, wall-surface-0..100, wall-back-0..100, wall-base-0..100, wall-tip-0..100

surface runs left to right across the floor; bottom, left, and right cover its other edges. wall-surface and wall-back run from root to tip; wall-base and wall-tip span the end caps. transition runs from pulley-center to wall-root. Upward walls require zero inset and drop. Use 50 for any edge midpoint.

10.3 platform (node)

Key	Default
minimum width	5cm
minimum height	2cm
strip width	0.25cm
wall angle	-90 degrees
left wall angle	-90 degrees
right wall angle	-90 degrees
wall inset	0cm
left wall inset	0cm
right wall inset	0cm
wall drop	0cm
left wall drop	0cm
right wall drop	0cm
pulley edge	off; shared preset uses 0.5cm inset and drop
left pulley edge	off; left-side preset uses 0.5cm inset and drop
right pulley edge	off; right-side preset uses 0.5cm inset and drop
pulley edges	off; both-side alias of pulley edge

Size aliases: platform width = minimum width; platform depth = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, surface, left-pulley-center, right-pulley-center, left-wall-root, right-wall-root, floor-center, left-wall-center, right-wall-center, text

Families: bottom-0..100, right-0..100, surface-0..100, left-0..100, left-transition-0..100, right-transition-0..100, left-wall-surface-0..100, left-wall-back-0..100, left-wall-base-0..100, left-wall-tip-0..100, right-wall-surface-0..100, right-wall-back-0..100, right-wall-base-0..100, right-wall-tip-0..100

surface runs left to right across the floor. Each wall has surface, back, base, and tip families with left-wall or right-wall prefixes; each transition runs from its pulley-center to wall-root. Upward walls require zero inset and drop. Use 50 for any edge midpoint.

10.4 wedge (node)

Key	Default
minimum width	4cm
wedge height	derived
wedge angle	30 degrees
wedge right angle at	br
wedge top inset	0cm
wedge top drop	0cm
pulley edge	off; preset uses 0.5cm inset and 1cm drop
surface guide length	0.5cm

Size aliases: wedge width = minimum width.

Aliases: surface-start, surface-end, center, north, south, east, west, north east, north west, south east, south west, bl, br, top, pulley-center, wall-root, transition-mid, base-mid, right-mid, slope-mid, slope-right-mid, centroid, mid, text

Families: base-0..100, right-0..100, slope-0..100, transition-0..100, surface-0..100, tangent-before-0..100, tangent-after-0..100, normal-0..100, right-surface-0..100, right-tangent-before-0..100, right-tangent-after-0..100, right-normal-0..100

base, right, and slope follow the outline counterclockwise from bl. surface runs left to right on the primary usable incline; in top mode right-surface selects the second incline. right-mid equals right-50; slope-mid equals surface-50. A pulley edge is supported in br and bl modes, and its centroid follows the filled four-point body. Directions and normals are local to the node.

10.5 ground (node)

Key	Default
minimum width	6cm
minimum height	0.3cm

Size aliases: ground width = minimum width; ground depth = minimum height.

Aliases: center, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right, surface, text

Families: bottom-0..100, right-0..100, surface-0..100, left-0..100

surface runs left to right on the top contact face; bottom, left, and right cover the other edges. Use 50 for an edge midpoint.

10.6 ceiling (node)

Key	Default
minimum width	2cm
minimum height	0.3cm

Size aliases: ceiling width = minimum width; ceiling depth = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right, surface, text

Families: surface-0..100, right-0..100, top-0..100, left-0..100

surface runs left to right on the underside; top, left, and right cover the other edges. Use 50 for an edge midpoint.

10.7 wall-left (node)

Key	Default
minimum width	0.3cm
minimum height	3cm

Size aliases: wall thickness = minimum width; wall height = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right, surface, text

Families: bottom-0..100, surface-0..100, top-0..100, left-0..100

surface runs bottom to top on the right contact face; top, bottom, and left cover the other edges. Use 50 for an edge midpoint.

10.8 wall-right (node)

Key	Default
minimum width	0.3cm
minimum height	3cm

Size aliases: wall thickness = minimum width; wall height = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right, surface, text

Families: bottom-0..100, right-0..100, top-0..100, surface-0..100

surface runs bottom to top on the left contact face; top, bottom, and right cover the other edges. Use 50 for an edge midpoint.

10.9 ramp (node)

Key	Default
minimum width	8.6cm
ramp run	2.6cm
ramp rise	derived
ramp angle	30
ramp depth	0.25cm
ramp wall height	1.5cm
ramp wall width	0.25cm
ramp direction	right
ramp guide length	0.5cm

Anchors: center, surface-start, surface-mid, surface-end, floor-start, floor-mid, floor-end, ramp-foot, ramp-mid, ramp-top, wall-top, wall-mid, wall-bottom, base-start, base-end, angle-baseline, angle-label, angle-first, angle-second, text

Families: floor-0..100, incline-0..100, wall-surface-0..100, wall-back-0..100, wall-tip-0..100, base-0..100, end-0..100, surface-0..100, tangent-before-0..100, tangent-after-0..100, normal-0..100; legacy .0..100 shorthand also available

surface follows floor plus incline by distance; floor and incline select each component. wall-surface, wall-back, wall-tip, base, and end cover the remaining outline. At the sharp joint the tangent uses the incline side.

10.10 curved-ramp (node)

Key	Default
curved ramp floor length	3.75cm
curved ramp radius	4cm
curved ramp angle	90
curved ramp back extension	1cm
ramp depth	0.25cm
ramp direction	right
ramp guide length	0.5cm

Anchors: center, surface-start, surface-mid, surface-end, curve-center, curve-start, curve-mid, curve-end, floor-start, floor-mid, floor-end, base-start, base-end, back-top, back-bottom, text

Families: floor-0..100, start-0..100, base-0..100, back-0..100, top-0..100, surface-0..100, curve-0..100, tangent-before-0..100, tangent-after-0..100, normal-0..100, curve-tangent-before-0..100, curve-tangent-after-0..100, curve-normal-0..100; legacy .0..100 shorthand also available

surface follows floor plus arc by distance; curve follows the arc only. floor, start, base, back, and top cover the remaining boundary.

10.11 block (node)

Key	Default
minimum width	1cm
minimum height	1cm

Size aliases: block width = minimum width; block height = minimum height.

Anchors: center, north, south, east, west, north east, north west, south east, south west, base, base east, base west, mid, mid east, mid west, text

Families: bottom-0..100, right-0..100, top-0..100, left-0..100; Bare numeric anchors remain angles in degrees, as in ordinary TikZ.

bottom, right, top, left follow the boundary counterclockwise. Use 0 for an edge start, 50 for its midpoint, 100 for its end.

10.12 pulley (node)

Key	Default
minimum size	1cm
pulley axle radius	1.5pt
pulley axle color	black

Size aliases: pulley diameter = minimum size.

Anchors: center, north, south, east, west, north east, north west, south east, south west, base, base east, base west, mid, mid east, mid west, text

Families: rim-0..100; Bare numeric anchors remain angles in degrees, as in ordinary TikZ.

rim starts at the rightmost point and runs counterclockwise around the circle. 25 is north, 50 west, 75 south, and 100 returns to the start.

10.13 particle (node)

Key	Default
minimum size	3mm

Anchors: center, north, south, east, west, north east, north west, south east, south west, base, base east, base west, mid, mid east, mid west, text

Families: rim-0..100; Bare numeric anchors remain angles in degrees, as in ordinary TikZ.

rim starts at the rightmost point and runs counterclockwise around the circle. 25 is north, 50 west, 75 south, and 100 returns to the start.

10.14 disk (node)

Key	Default
minimum size	15mm

Anchors: center, north, south, east, west, north east, north west, south east, south west, base, base east, base west, mid, mid east, mid west, text

Families: rim-0..100; Bare numeric anchors remain angles in degrees, as in ordinary TikZ.

rim starts at the rightmost point and runs counterclockwise around the circle. 25 is north, 50 west, 75 south, and 100 returns to the start.

10.15 ring (node)

Key	Default
minimum size	15mm

Anchors: center, north, south, east, west, north east, north west, south east, south west, base, base east, base west, mid, mid east, mid west, text

Families: rim-0..100; Bare numeric anchors remain angles in degrees, as in ordinary TikZ.

rim starts at the rightmost point and runs counterclockwise around the circle. 25 is north, 50 west, 75 south, and 100 returns to the start.

10.16 spring (node and path)

Key	Default
minimum width	3cm (node)
minimum height	0pt (node)
pre length	5pt
post length	5pt
amplitude	4.5pt
segment length	4.5pt
aspect	0.5

Size aliases: spring length = minimum width.

Anchors: center, start, end, coil-start, coil-end, north, south, east, west, north east, south west, north west, south east, text

Families: axis-0..100

Use `draw[spring] (A)–(B)` for a two-point connection, or `node[spring] (S)` for a positioned coil. Node length defaults to 3cm; spring length and minimum width set its length. start/end are exact axis attachments, coil-start/coil-end delimit the leads, and axis-0..100 samples the straight axis, not the coil wire. show anchors/show keys apply to nodes; physicshelpspring describes both forms. Lead lengths may be zero; their sum must be smaller than the node length. amplitude and segment length must be positive.

10.17 rope (path)

Key	Default
line width	0.5pt
over pulley	required node name
string route	surface-right
string start solution	2
string end solution	1
string wrap	clockwise

Anchors: No private anchors; use path endpoints and nodes along the path.

Use `draw[rope] (A) to[over pulley=P] (B)`. Endpoints must lie outside a circular pulley. Labels on to sit on the final straight segment. shortest changes wrap only, not the tangent pair.

10.18 force (path)

Key	Default
line width	0.7pt

anchors: No private anchors; use path endpoints and nodes along the path.

10.19 velocity (path)

Key	Default
line width	0.6pt

anchors: No private anchors; use path endpoints and nodes along the path.

10.20 acceleration (path)

Key	Default
line width	0.6pt

anchors: No private anchors; use path endpoints and nodes along the path.

10.21 torque (path)

Key	Default
line width	0.7pt

anchors: No private anchors; use path endpoints and nodes along the path.

10.22 rod (path)

Key	Default
line width	2pt

anchors: No private anchors; use path endpoints and nodes along the path.

10.23 pin-support (pic)

Key	Default
support size	6mm

anchors: pivot, base, left, right

Use pic (S) pin-support; anchors are coordinates (S-pivot), (S-base), (S-left), (S-right).

10.24 roller-support (pic)

Key	Default
support size	6mm

anchors: pivot, base, left, right

Use pic (S) roller-support; anchors are coordinates (S-pivot), (S-base), (S-left), (S-right).

10.25 pendulum (pic)

Key	Default
pendulum length	2cm
pendulum angle	30 degrees

Anchors: pivot, bob (node)

Use pic (P) pendulum. Angle is measured from downward vertical; positive swings right. Length is pivot to bob centre. Use (P-pivot) and (P-bob.center).

10.26 polar element (node)

Key	Default
element inner radius	1.8cm
element outer radius	2cm
element radial thickness	derived
element start angle	30 degrees
element delta angle	30 degrees
element dimension offset	4mm
element radius label	r
element radial label	dr
element angular label	d theta
element arc label	r d theta
show dimensions	off

Anchors: center, text, north, south, east, west, north east, north west, south east, south west, centroid, inner-start, inner-mid, inner-end, outer-start, outer-mid, outer-end, start-mid, end-mid

Families: inner-0..100, outer-0..100, start-0..100, end-0..100

An annular sector from inner-start to outer-end. Set inner radius to zero for a sector or delta angle to 360 degrees for a full ring.

10.27 differential sector (node)

Key	Default
element inner radius	0cm
element outer radius	2cm
element radial thickness	derived
element start angle	30 degrees
element delta angle	30 degrees
element dimension offset	4mm
element radius label	r
element angular label	d theta
element arc label	r d theta
show dimensions	off

Anchors: center, text, north, south, east, west, north east, north west, south east, south west, centroid, inner-start, inner-mid, inner-end, outer-start, outer-mid, outer-end, start-mid, end-mid

Families: inner-0..100, outer-0..100, start-0..100, end-0..100

A polar-element preset with zero inner radius. The outer arc represents r d theta and numeric families follow every boundary.

10.28 differential ring (node)

Key	Default
element inner radius	1.8cm
element outer radius	2cm
element radial thickness	derived
element delta angle	360 degrees
element radius label	r
element radial label	dr
show dimensions	off

Anchors: center, text, north, south, east, west, north east, north west, south east, south west, centroid, inner-start, inner-mid, inner-end, outer-start, outer-mid, outer-end, start-mid, end-mid

Families: inner-0..100, outer-0..100, start-0..100, end-0..100

A polar-element preset with a 360-degree sweep. Use source element on an unwrapped ring to copy its radii.

10.29 unwrapped ring (node)

Key	Default
element inner radius	1.8cm
element outer radius	2cm
element radial thickness	derived
source element	optional named element
element circumference label	$2\pi r$
element radial label	dr
show dimensions	off

Anchors: center, text, centroid, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right

Families: bottom-0..100, right-0..100, top-0..100, left-0..100

A differential approximation with width 2π times the inner reference radius and height equal to the radial thickness. source element copies both radii from a named polar element.

10.30 spherical shell (node)

Key	Default
element inner radius	1cm
element outer radius	derived
element radial thickness	1.5mm
element delta angle	360 degrees
element radius label	r
element radial label	dr
show dimensions	off

Anchors: center, text, north, south, east, west, north east, north west, south east, south west, centroid, inner-start, inner-mid, inner-end, outer-start, outer-mid, outer-end, start-mid, end-mid

Families: inner-0..100, outer-0..100, start-0..100, end-0..100

A concentric differential shell shown in section. Its full-sweep anchors and radial geometry match the spherical-shell construction.

10.31 hollow sphere (node)

Key	Default
element inner radius	1cm
element outer radius	2cm
element radial thickness	derived
element delta angle	360 degrees
element cavity radius label	a
element body radius label	R
show dimensions	off

Anchors: center, text, north, south, east, west, north east, north west, south east, south west, centroid, inner-start, inner-mid, inner-end, outer-start, outer-mid, outer-end, start-mid, end-mid

Families: inner-0..100, outer-0..100, start-0..100, end-0..100

A finite spherical wall shown in section, with independent inner and outer radii.

10.32 rectangular element (node)

Key	Default
element width	8mm
element height	6mm
element width label	dx
element height label	dy
show dimensions	off

Anchors: center, text, centroid, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right

Families: bottom-0..100, right-0..100, top-0..100, left-0..100

A Cartesian differential area with independent width and height, complete boundary anchors, and optional dimension labels.

10.33 rectangular strip (node)

Key	Default
element width	4cm
element height	2mm
element width label	b
element height label	dx
show dimensions	off

Anchors: center, text, centroid, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right

Families: bottom-0..100, right-0..100, top-0..100, left-0..100

A full-width strip preset. Change its width and differential thickness independently.

10.34 rectangular sheet (node)

Key	Default
element width	5cm
element height	3cm

Aliases: center, text, centroid, north, south, east, west, north east, north west, south east, south west, top-left, top-right, bottom-left, bottom-right

Families: bottom-0..100, right-0..100, top-0..100, left-0..100

A finite unshaded sheet node used alone or as the body in a sheet element diagram.

10.35 differential ring diagram (pic)

Key	Default
element inner radius	1cm
element outer radius	1.2cm
element radial thickness	derived
element body radius	2cm
element diagram gap	5mm
element center dot radius	1.2pt
show dimensions	on for both parts

Aliases: center, body-north, body-south, body-east, body-west, ring (node), strip (node)

Use pic (D) differential ring diagram=.... It keeps the body circle, ring, centre mark, and unwrapped strip together. Components are (D-ring) and (D-strip).

10.36 sphere shell diagram (pic)

Key	Default
element body radius	2cm
element inner radius	1cm
element radial thickness	1.5mm
element formula label	$dV = 4 \pi r^2 dr$

Aliases: center, body (node), shell (node), formula-anchor

A solid sphere with a highlighted spherical shell. Components are (S-body) and (S-shell); (S-center) marks the common centre.

10.37 sphere slice diagram (pic)

Key	Default
element body radius	2cm
element axial thickness	2mm
element position	0.35 in (-1
element projection ratio	0.24
element axial label	dx
element formula label	$dV = \pi y^2 dx$

Aliases: center, body (node), slice (node), formula-anchor

A solid sphere with a projected disk slice at a signed normalized axial position. Components are (S-body) and (S-slice).

10.38 hollow sphere diagram (pic)

Key	Default
element inner radius	9mm
element outer radius	2cm
element cavity radius label	a
element body radius label	R
element formula label	$V = 4 \pi \text{ over 3 times } (R^3 - a^3)$

anchors: center, body-north, body-south, wall (node), formula-anchor

A hollow sphere in section. Component (H-wall) exposes the inner and outer boundaries.

10.39 cylinder shell diagram (pic)

Key	Default
element body radius	1.6cm
element body height	3.4cm
element inner radius	9mm
element radial thickness	1.5mm
element projection ratio	0.25
element formula label	$dV = 2 \pi r h dr$

anchors: center, body (node), shell or slice (node), formula-anchor

A projected cylinder with a radial cylindrical shell. Components are (C-body) and (C-shell).

10.40 cylinder slice diagram (pic)

Key	Default
element body radius	1.6cm
element body height	3.4cm
element axial thickness	2mm
element position	0.55
element projection ratio	0.25
element axial label	dz
element formula label	$dV = \pi R^2 dz$

anchors: center, body (node), shell or slice (node), formula-anchor

A projected cylinder with an axial disk slice. Components are (C-body) and (C-slice).

10.41 cone slice diagram (pic)

Key	Default
element body radius	1.8cm
element body height	4cm
element axial thickness	2mm
element position	0.58 from apex
element projection ratio	0.24
element axial label	dx
element formula label	$dV = \pi r(x)^2 dx$

anchors: center, apex, base, body (node), slice (node), formula-anchor

A projected cone with a thin disk slice. element position is measured from apex to base. Components are (K-body) and (K-slice).

10.42 sheet element diagram (pic)

Key	Default
element body width	5cm
element body height	3cm
element width	8mm
element height	6mm
element position	0.58
element transverse position	0.58
element formula label	$dA = dx dy$

anchors: center, body (node), element (node), formula-anchor

A rectangular sheet with a named Cartesian element. Components are (A-body) and (A-element).

10.43 u tube diagram (pic)

Key	Default
fluid width	2cm
fluid height	4cm
fluid tube width	.3cm
fluid left level	.75
fluid right level	.55
fluid level	.65

anchors: origin, center, bottom, top, left, right, left-surface, right-surface, interface, upper-surface

Two-liquid U tube. level adds a second liquid only above left level. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.44 gas manometer (pic)

Key	Default
fluid width	1.8cm
fluid height	3.8cm
fluid tube width	.3cm
fluid left level	.45
fluid right level	.8

Anchors: origin, center, bottom, top, left, right, left-surface, right-surface, gas

Gas reservoir connected to a U tube. body label names the gas. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.45 hydraulic press (pic)

Key	Default
fluid width	4cm
fluid height	3.5cm
fluid tube width	.3cm
fluid left level	.4
fluid right level	.8

Anchors: origin, center, bottom, top, left, right, large-piston, small-piston

Large piston width is 0.32 times width; tube width sets the small piston. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.46 fluid cylinder diagram (pic)

Key	Default
fluid width	1.6cm
fluid height	3.2cm
fluid level	.65

Anchors: origin, center, bottom, top, left, right, surface

Projected cylinder with an elliptical liquid surface. width is the diameter. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.47 fluid tank diagram (pic)

Key	Default
fluid width	3cm
fluid height	3cm
fluid left level	.7
fluid right level	.7

Anchors: origin, center, bottom, top, left, right, left-surface, right-surface, surface

Open vessel. Equal endpoint levels give a horizontal free surface. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.48 pressure element diagram (pic)

Key	Default
fluid width	2cm
fluid height	.65cm

Anchors: origin, center, bottom, top, left, right

Projected horizontal fluid element with inward pressure forces. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.49 rotating tube (pic)

Key	Default
fluid width	5cm
fluid height	.5cm
fluid tube width	.3cm
fluid level	.48

anchors: origin, center, bottom, top, left, right, element-left, element-right

Horizontal tube with a differential section. level locates its left face along width. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.50 rotating fluid diagram (pic)

Key	Default
fluid width	3cm
fluid height	3.5cm
fluid level	.5
fluid bend	.35

anchors: origin, center, bottom, top, left, right, vertex, rim-surface

Parabolic surface. level is the vertex height; bend is the wall-to-vertex rise divided by height. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.51 buoyancy (pic)

Key	Default
fluid width	3cm
fluid height	3cm
fluid left level	.8
fluid right level	.8

anchors: origin, center, bottom, top, left, right, left-surface, right-surface, surface, body

Immersed body with schematic inward pressure arrows. Arrow lengths do not encode pressure magnitude. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.52 flow tube diagram (pic)

Key	Default
fluid width	5cm
fluid height	1.3cm
fluid bend	.45

anchors: origin, center, bottom, top, left, right, inlet, outlet

Curved flow tube; outlet height is 0.55 times inlet height. bend controls vertical curvature. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.53 meniscus diagram (pic)

Key	Default
fluid width	1.2cm
fluid height	2.7cm
fluid level	.7
fluid bend	.2

Anchors: origin, center, bottom, top, left, right, contact-left, contact-right, surface

Positive bend is concave; negative bend is convex. Contact angle is measured through the liquid. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.54 capillary (pic)

Key	Default
fluid width	4cm
fluid height	3cm
fluid level	.35
fluid left level	.7
fluid bend	.12

Anchors: origin, center, bottom, top, left, right, contact-left, contact-right, reservoir

Capillary rise. level is the reservoir surface and left level is the contact-line height. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.55 surface tension ring (pic)

Key	Default
fluid width	3cm
fluid height	1cm

Anchors: origin, center, bottom, top, left, right

Projected circular contact line. width and height are the ellipse diameters. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.56 liquid ring diagram (pic)

Key	Default
fluid width	3.5cm
fluid tube width	.4cm
fluid angle	225
fluid sweep	70

Anchors: origin, center, bottom, top, left, right, liquid-start, liquid-end

Annular liquid sector. width is outer diameter, tube width is radial thickness. Angles use degrees counter-clockwise from right. Use (F-origin), (F-center), and the listed (F-...) coordinates; these are pic coordinates, not node anchors.

10.57 fluid tank (node)

Key	Default
fluid width	3cm
fluid height	3cm
fluid left level	.7
fluid right level	.7

Anchors: center, north, south, east, west, north east, north west, south east, south west, left-surface, right-surface, surface, text, base, base east, base west, mid, mid east, mid west

Families: surface-0..100, bottom-0..100, right-0..100, left-0..100, bottom-left-0..100, bottom-right-0..100

surface runs left to right. bottom, right, left and the two rounded bottom corners follow the drawn outline counterclockwise.

10.58 meniscus (node)

Key	Default
fluid width	1.2cm
fluid height	2.7cm
fluid level	.7
fluid bend	.2

Anchors: center, north, south, east, west, north east, north west, south east, south west, left-surface, right-surface, surface, text, base, base east, base west, mid, mid east, mid west

Families: surface-0..100, bottom-0..100, right-0..100, left-0..100

surface follows the cubic liquid boundary from left to right. bottom, right and left follow the vessel boundary counterclockwise.

10.59 rotating fluid (node)

Key	Default
fluid width	3cm
fluid height	3.5cm
fluid level	.5
fluid bend	.35

Anchors: center, north, south, east, west, north east, north west, south east, south west, left-surface, right-surface, surface, text, base, base east, base west, mid, mid east, mid west

Families: surface-0..100, bottom-0..100, right-0..100, left-0..100

surface follows the parabola from left to right. bottom, right and left follow the vessel boundary counterclockwise.

10.60 fluid cylinder (node)

Key	Default
fluid width	1.6cm
fluid height	3.2cm
fluid level	.65

Anchors: center, north, south, east, west, north east, north west, south east, south west, left-surface, right-surface, surface, text, base, base east, base west, mid, mid east, mid west

Families: surface-0..100, surface-rim-0..100, top-0..100, bottom-0..100, right-0..100, left-0..100

surface is the left-to-right diameter of the liquid ellipse; surface-rim, top and bottom are full ellipse boundaries, starting at the rightmost point and running counterclockwise. left and right cover the straight walls.

10.61 pressure element (node)

Key	Default
fluid width	2cm
fluid height	.65cm

Anchors: center, north, south, east, west, north east, north west, south east, south west, inlet, outlet, text, base, base east, base west, mid, mid east, mid west

Families: inlet-0..100, outlet-0..100, inlet-rim-0..100, outlet-rim-0..100, axis-0..100, top-0..100, bottom-0..100

inlet and outlet run bottom to top across their projected faces. inlet-rim and outlet-rim run counterclockwise from the rightmost point. axis joins their centres left to right; top and bottom cover the straight outline.

10.62 flow tube (node)

Key	Default
fluid width	5cm
fluid height	1.3cm
fluid bend	.45

Anchors: center, north, south, east, west, north east, north west, south east, south west, inlet, outlet, text, base, base east, base west, mid, mid east, mid west

Families: inlet-0..100, outlet-0..100, axis-0..100, bottom-0..100, top-0..100

inlet and outlet run bottom to top; axis follows their curved midline left to right. bottom runs left to right, top right to left on the actual cubic outline. Curve percentages use the cubic parameter.

10.63 liquid ring (node)

Key	Default
fluid width	3.5cm
fluid tube width	.4cm
fluid angle	225
fluid sweep	70

Anchors: center, north, south, east, west, north east, north west, south east, south west, liquid-start, liquid-end, text, base, base east, base west, mid, mid east, mid west

Families: outer-0..100, inner-0..100, liquid-0..100, start-0..100, end-0..100

outer and inner run counterclockwise from the rightmost point. liquid follows the filled sector centreline from liquid-start to liquid-end; start and end run radially from inner to outer boundary.

10.64 u tube (node)

Key	Default
fluid width	2cm
fluid height	4cm
fluid tube width	.3cm
fluid left level	.75
fluid right level	.55

Anchors: center, north, south, east, west, north east, north west, south east, south west, left-surface, right-surface, surface, text, base, base east, base west, mid, mid east, mid west

Families: surface-0..100, left-surface-0..100, right-surface-0..100, outer-0..100, inner-0..100

left-surface and right-surface independently run left to right across the liquid in each arm. surface selects the left arm, never the empty gap. outer and inner follow the U outline by distance from the left lip down through the bend to the right lip.

10.65 concave-mirror (node)

Key	Default
mirror radius	5cm
mirror thickness	0.25cm
mirror aperture angle	30 degrees

Anchors: center, north, south, east, west, north east, north west, south east, south west, vertex, arc-mid, surface-mid, front-mid, back-mid, front-top, front-bottom, surface-top, surface-bottom, back-top, back-bottom, top, bottom, text

Families: surface-0..100, front-0..100, back-0..100; legacy .0..100 shorthand also available

10.66 convex-mirror (node)

Key	Default
mirror radius	5cm
mirror thickness	0.25cm
mirror aperture angle	30 degrees

Anchors: center, north, south, east, west, north east, north west, south east, south west, vertex, arc-mid, surface-mid, front-mid, back-mid, front-top, front-bottom, surface-top, surface-bottom, back-top, back-bottom, top, bottom, text

Families: surface-0..100, front-0..100, back-0..100; legacy .0..100 shorthand also available

10.67 convex-lens (node)

Key	Default
convex lens radius	5cm
convex lens thickness	0.2cm
convex lens aperture angle	25 degrees

Anchors: center, north, south, east, west, north east, north west, south east, south west, front-mid, back-mid, front-top, front-bottom, back-top, back-bottom, top, bottom, text

Families: front-0..100, back-0..100; legacy .0..100 shorthand also available

10.68 concave-lens (node)

Key	Default
concave lens radius	5cm
concave lens thickness	0.2cm
concave lens aperture angle	25 degrees

Aliases: center, north, south, east, west, north east, north west, south east, south west, front-mid, back-mid, front-top, front-bottom, back-top, back-bottom, top, bottom, text

Families: front-0..100, back-0..100; legacy .0..100 shorthand also available

10.69 slab (node)

Key	Default
minimum width	1.2cm
minimum height	3cm

Size aliases: slab width = minimum width; slab height = minimum height.

Aliases: center, north, south, east, west, north east, north west, south east, south west, front, back, front-top, front-bottom, back-top, back-bottom, top, bottom, text

Families: front-0..100, back-0..100, top-0..100, bottom-0..100; legacy .0..100 shorthand also available

10.70 prism (node)

Key	Default
minimum width	3cm
minimum height	2.6cm
prism apex angle	derived

Size aliases: prism width = minimum width; prism height = minimum height.

Aliases: center, centroid, north, south, east, west, north east, north west, south east, south west, apex, base-left, base-right, base-mid, left-mid, right-mid, entry-mid, exit-mid, text

Families: base-0..100, right-0..100, left-0..100; legacy .0..100 shorthand also available

11 License

Copyright © 2026 Vaibhav Blayer.

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3c or (at your option) any later version. This work has LPPL maintenance status “maintained”. The Current Maintainer is Vaibhav Blayer.