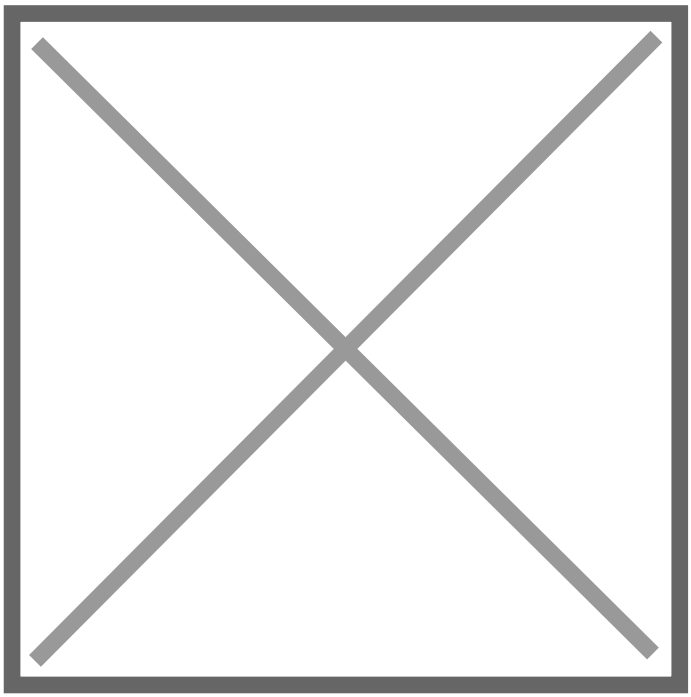
MIDILLI SVG Generator

A free and versatile command-line tool for creating SVG files to prepare box-shaped parts using Laser Engravers/Cutters.

- Introduction
- Features
- How does it work?
- Download
- Version History
- Quick Usage
- Coordinate system and effects of sizing parameters
- Command line arguments
- JSON File
- Typical scenario for separate files
- Additional Resources

Introduction

MIDILLI SVG Generator is a free and versatile command-line tool for creating SVG files to prepare box-shaped parts. It supports circular drill holes, text, rectangular cutouts, and precise dimensions in metric units. The generated SVG files are compatible with laser cutters and engravers, ensuring seamless design preparation.



The final result of the simple example project given in this manual.

Derived from the in-house MIDILLI Tech Automation Suite, this tool is a simplified yet robust solution.

It is offered free for non-commercial use. You can find the full license details here.

> Warning: Operating laser engravers or cutters requires a safe environment to prevent exposure to toxic gases and strong laser light that could harm your eyes. Always ensure proper ventilation and wear safety glasses, and follow material guidelines. Never engrave prohibited materials, which can release harmful fumes.

Please choose a topic from left to read.

Features

- **Box Creation**:
  - Generate rectangular shapes with up to six sides to form a complete box structures.
  - Option to omit any side during box design.
  - Option to use the sizing for the inner volume or the outer volume.
- **Customizable Cutouts**:
  - Create holes or rectangular cutouts (also called components) on specific sides of the box with precise locations.
  - Create screw holes on top or bottom* side with the given screw diameter.
- **Component Rotation**:
  - Rotate components to any specified angle.
- **Text Customization**:
  - Add text to components with customizable font sizes.
  - Text alignment follows the rotation of the component.
  - Maintain visual margins between text and component edges.
- **Recess Support:**
  - Create flush surfaces with custom dimensions that prevent movement of e.g., button caps and improve functionality.
  - Create recesses for screw heads*.
- **Laser Kerf Compensation:**
  - Automatically adjust design if your laser software doesn’t support kerf.
- **Separate Output Files for each Layer**:
  - Generate distinct files for outlines, components, text, screws, and recesses for different purposes, such as engraving and cutting with different settings.
- **Vectorization**:
  - Convert SVGs into vectorized format using Inkscape (needs the free and open-source Inkspace installed separately).
- **Material Thickness**:
  - Adjust material thickness for accurate cuts and fits.
- **Box Sizing**:
  - Automatically adjust box sizes for inner or outer edges, depending on user preference.

* Using recess for screws on bottom side requires manual flipping of the surface.

How does it work?

The tool takes a JSON file as input (example below) and generates SVG files for up to six sides of a box, with each side featuring its own drill cutouts. The pieces can then be combined together to construct a 3D shape.

If Inkscape is installed, you can vectorize the SVG files directly without opening them, making them ready for use with your laser software.

Download

By downloading our free software, you agree to the license conditions.

Download (Windows, Linux, macOS)

Version History

- 1.4.0 (27.08.2026):
  - Added finger joint support with `--finger-joint` and per-side `--no-finger-*` options.
  - Fixed finger joint default behavior and effective settings display.
- 1.3.0 (05.11.2025):
  - Now available for Windows, Linux, and macOS as well.
  - `box-height-3d`, `material-thickness` and `outer-sizing` arguments can be also used in JSON (can be overridden via command-line argument)
- 1.2.0 (23.05.2025):
  - Added laser kerf support for compensation.
- 1.1.2 (01.04.2025):
  - Minor bug fixes.
- 1.1.1 (08.02.2025):
  - Replaced `height` with `length` in JSON to avoid confusion.
- 1.1.0 (07.02.2025):
  - No more angle limitation for text rotation.
  - Added adjustable font sizes for each text.
- 1.0.0 (22.01.2025):
  - Initial version.

Quick Usage

Download the ZIP file and extract to a location. Copy the *Example JSON file* below and save it as *test.json* using a simple text editor. To keep the things simple, save it at the same location as the executable.

To create your own JSON file, you can use our visual box designer, our online JSON Creator or refer to MIDILLI JSON Definition document to create your own from scratch.

Example JSON File

The following file generates all six sides for *MyPart*, placing one hole and one rectangle cutout on the *Top* side. The top size of the box is 5x5 cm and the components are placed at (x = 1, y = 1) and (x = 3, y = 3) respectively. The visible sizes of the components are the same as drill sizes and their texts are just placed above since they are zero degrees rotated, with a font size of 8 pixels.

```
{
  "Generic": {
    "width": 5.0,
    "length": 5.0
  },
  "MyPart_Top": {
    "myRoundComponent#1": {
      "location_x": 1,
      "location_y": 1,
      "drill_diameter": 1,
      "call_name": "Hello",
                  "visible_diameter": 0,
      "visible_length" : 1,
      "visible_width" : 1,
      "rotation": 0
    },
    "myRectangleComponent#1": {
      "location_x": 3,
      "location_y": 3,
      "drill_width": 2,
      "drill_length": 1,
```

```
□□□"call_name": "MIDILLI",
□□□"visible_width" : 1,
□□□"visible_length" : 2,
□□□"rotation": 0
□□}
□},
□"MyPart_Bottom": {

□},
□"MyPart_Back": {
□□
□},
□"MyPart_Front": {

□},
□"MyPart_Left": {

□},
□"MyPart_Right": {

□}
}
```

Depending on the command-line options, the other sides of the part will be created using different sizing. Read *Coordinate system and effect of sizing parameters* for more on different results of the arguments.

> **Note**: JSON syntax is very strict. Ensure that the last line before each '}' does **not** end with a comma, while all other lines **should**. Use an online validation tool to confirm correctness.

> **Note**: Use dot (.) if working with decimal numbers to separate the integer part.

Assuming that the JSON file is stored at the same location as the executable file, open a command window (Command Prompt, or PowerShell), then execute the following command:

```
.\SVGGenerator.exe --json-file .\test.json --output-directory . --material-thickness 3 --box-height-3d 30 --vectorize=false
```

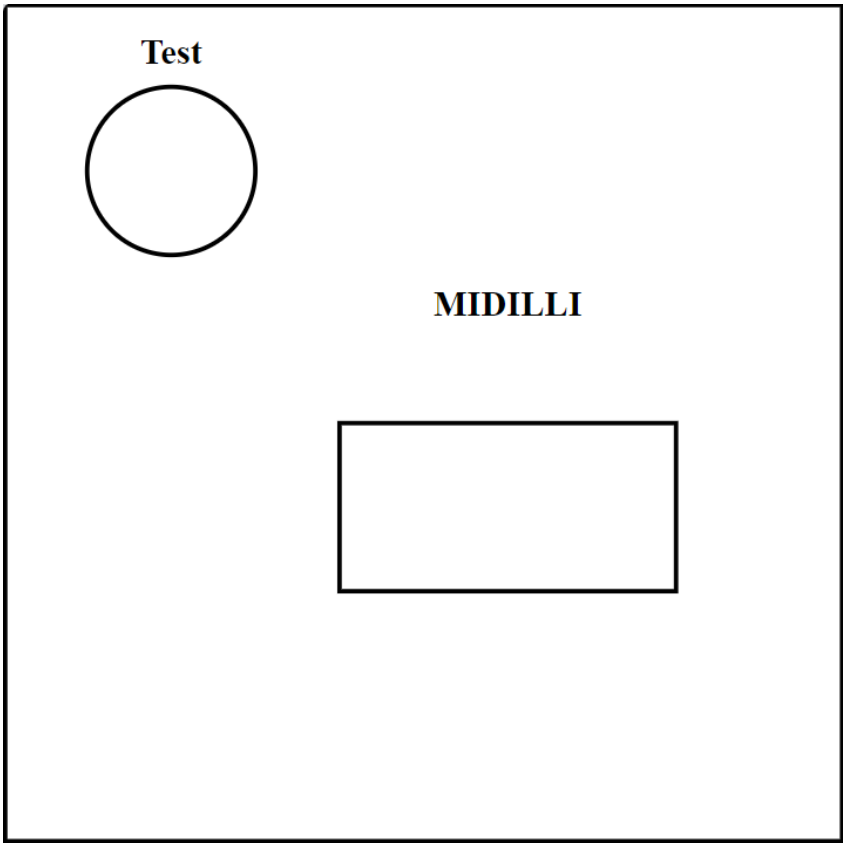
You will be provided with an output showing the current settings as following:

```
~~~~~ Settings ~~~~~
Laser kerf: 0 mm
Box height in 3D: 30 mm
Outer sizing enabled: No
Finger joints: No
Finger size: 3 mm (default: material thickness)
Finger count: Automatic
Finger clearance: 0 mm
Text enabled: Yes (Default font size: 8 px)
Components enabled: Yes
Side outline enabled: Yes
Recess for components enabled: No
Line per mm for recess areas: 1
Material thickness: 3 mm
Screw size top: 0 mm
Screw size bottom: 0 mm
Screw head diameter: 0 mm
Screw extra margin: 0 mm
Separate all output files: No
Vectorize enabled: No
Keep original files: No
Simulation enabled: No
Inkscape path: C:\\Progra~1\\Inkscape\\bin\\inkscape.com
```

Each setting on the console output can be customized with command line arguments.

Note: Use dot (.) if working with decimal numbers to separate the integer part.

After completion, 6 files starting with *MyPart_* and ending with *.svg* will be visible under *test* directory. Opening *MyPart_Top.svg* should display the following:



If you would like to directly vectorize the output, you can remove `--vectorize=false` option. As stated above, this will require the *Inkscape* to be installed.

For more customization, read the *Detailed usage* section.

Hint

Tip: If you want the overall size to be exactly as you specify in the JSON file, use the --outer-sizing option. If you want the usable inner area to be the same as you define, leave it as default.

Coordinate system and effects of sizing parameters

Read more on the following page: Understanding 2D Coordinate Systems in MIDILLI Tools.

Command line arguments

Introduction

If the program is executed with `-h` or `--help` option, the following quick help will be displayed:

```
Usage:
  SVGGenerator.exe [OPTION...]

  -j, --json-file arg           Path to the project JSON file
  -o, --output-directory arg    Directory to store the output
  -q, --kerf-width arg          Laser kerf in mm (default: 0)
  -m, --material-thickness arg  Material thickness in mm (default: 0)
  -b, --box-height-3d arg       Box height in 3D in mm (default: 0)
  -t, --text                    Enable text output of components (default:
                                true)
  -f, --font-size arg           Default font size of component text in px
                                (default: 8)
  -c, --components              Enable component output (default: true)
  -l, --outline                 Enable side outline output (default: true)
  -r, --recess                  Enable recess output for components
  -n, --recess-lines-per-mm arg
                                Recess lines per mm (default: 1)
  -v, --vectorize               Vectorize output (default: true)
  -e, --separate-files          Separate all outputs to their files for
                                easier engraving and cutting
  -k, --keep-files              Keep non-vectorized files
  -z, --outer-sizing            Box sizes are for outside borders (inner
                                area will be smaller)
      --finger-joint            Add finger joints (box joints) to all 6 sides
                                so panels can be assembled without glue.
                                When enabled all 6 sides default on; use
                                --no-finger-<side> to disable a specific side
      --finger-size arg         Finger width in mm (0 = material thickness)
                                (default: 0)
```

```
      --finger-count arg        Number of fingers on an edge (0 = automatic
                                from finger size) (default: 0)
      --finger-clearance arg    Extra clearance added to finger slots in mm
                                (default: 0)
      --no-finger-top           Disable finger joints on the Top panel
      --no-finger-bottom        Disable finger joints on the Bottom panel
      --no-finger-front         Disable finger joints on the Front panel
      --no-finger-back          Disable finger joints on the Back panel
      --no-finger-left          Disable finger joints on the Left panel
      --no-finger-right         Disable finger joints on the Right panel
  -p, --screw-size-top arg      Screw diameter top in mm (default: 0)
  -w, --screw-size-bottom arg   Screw diameter bottom in mm (default: 0)
  -d, --screw-head-diameter arg
                                Screw head diameter in mm (default: 0)
  -x, --screw-extra-margin arg  Screw extra margin in mm (default: 0)
  -s, --simulate                Simulate, do not write any files
  -i, --inkscape-path arg       Inkscape command (default:
                                C:\\Progra~1\\Inkscape\\bin\\inkscape.com)
  -h, --help                    Show this help and quit
```

Both short and long versions of the arguments can be used.

The arguments `--json-file`, `--output-directory`, `--material-thickness`, and `--box-height-3d` are mandatory. The last two arguments can also be given in the JSON file. If both JSON and the command-line has these arguments, the command line values will be used. Other arguments have default values as mentioned above. To set an argument to false, use the long version and assign the value `false`. For example, to disable text output, use `--text=false`. Similarly, for activation od options, which are disabled by default, only use the argument. For example, to enable simulation, use either `-s` or `--simulate`. Argument values can be provided with a space or an equal sign if long versions are used. For instance, both `--json-file test.json` and `--json-file=test.json` are valid usage. For short versions, only space allowed, e.g., `-b 30`.

`--json-file`

A valid path to JSON file to be used as input. Can be a relative path (starting from the working directory) or an absolute path. You can use online visual box designer, JSON Creator or refer to MIDILLI JSON Definition document if you need help.

`--output-directory`

The directory to write the output files to. Using `.` (dot) will use the current working directory. The final files will be written into a new folder having the name of the `--json-file`.

`--kerf-width`

If your laser cutting software does not automatically compensate for the laser kerf (the material removed by the laser beam width), you can manually set the kerf value in the settings. This allows you to adjust your design dimensions so that the final cut parts have accurate sizes, ensuring a precise fit and clean finish.

Read this page to learn more about the effect of this parameter.

Default: 0 mm (no compensation)

`--material-thickness`

This value is used to determine the dimensions of the left, right, front, and bottom sides (if enabled) based on whether the `--outer-sizing` parameter is active. See *Coordinate system and effects of sizing parameters* for more info. The value is in millimeters (mm).

This value is mandatory.
It can be given in the JSON (under Generic, as value "`material_thickness`") as well as in the command line. Command line will override the value if provided.

`--box-height-3d`

The height of the box in millimeters (mm). If `--outer-sizing` is used, this will be the outer height of the box. Otherwise, it is the inner height of the box.

This value is mandatory.
It can be given in the JSON (under Generic, as value "`box_height_3d`") as well as in the command line. Command line will override the value if provided.

`--text`

Enables the text output of the components if they have a *call_name* property. `--font-size` argument specifies the size of the text. The rotation follows the *rotation* property of the component. Text rotates only for 0 90 180 270 degrees. Otherwise 0 (zero) is assumed.

Default: true

`--font-size`

Specifies the font size if the `--text` output is enabled. The value is in pixels (px). This value is used, if `silkscreen_font_size` property is not set in the JSON.

Default: 8 px

`--components`

Enables the component output to cut circles and/or rectangular shapes through the `--material-thickness`.

Default: true

`--outline`

Enables the side outline output which draws the sizes of the sides with the given dimensions based on the `--outer-sizing` property.

Default: true

`--recess`

Enables the recess output of the components. Recesses take the sizing from *visible_diameter* (priority) or *visible_width* and *visible_height* properties. Recesses may be helpful to have slightly lowered shapes to e.g., stabilize or limit the movement of button caps. This value does not change the behavior of the screw head recess (See `--screw-head-height`).

Default: false (do not create recesses for components)

`--recess-lines-per-mm`

If `--recess` is used, this argument specifies how many lines per millimeter (mm) should be drawn to create the recesses. This is important to set if the default value is not enough for the laser to create the recess.

This value is used both for `--recess` and `--screw-head-height` arguments.

Default: 1 line per mm

`--vectorize`

Vectorizes the SVG files using Inkscape (needs to be installed separately) in case the laser software is not capable of vectorizing the SVG files. Vectorizing files may take time for each side depending on the number of components on that side.

Default: true

`--separate-files`

Creates separate SVG files for each enabled output, by splitting them into different layers: `--components`, `--outline`, `--text`, `--recess`, `--screws-*` (for top and/or bottom), and `--screw-head-diameter` (screw head recesses for top and/or bottom). This is useful in case the laser software does not allow selections to specify different engraving/cutting settings. Each file will only contain the relevant layer.

Default: false, one file for each side, all layers merged.

`--keep-files`

Does not delete non-vectorized SVG files if `--vectorize` is enabled. An SVG editor can be used to view or modify the output.

Default: false, delete non-vectorized files if vectorization is successful.

`--outer-sizing`

If used, the sizes defined in the JSON file as well as the `--material-thickness` and `--box-height-3d` arguments will be used to create the outer volume with the given dimensions. Otherwise, these values will be used to create the inner volume with the given dimensions.

The value can be given in the JSON (under Generic, as value "`outer_sizing`") as well as in the command line. Command line will override the value if provided.

Default: false, use inner dimensions.

`--screw-size-top`

Screw hole diameter to be drawn as hole in millimeters (mm) on the top side of the box..

Default: 0 mm, no screws.

Brown: Screw diameter (or screw size)

`--screw-size-bottom`

Screw hole diameter to be drawn as hole in millimeters (mm) on the bottom side of the box.

Default: 0 mm, no screws.

Brown: Screw diameter (or screw size)

`--screw-head-diameter`

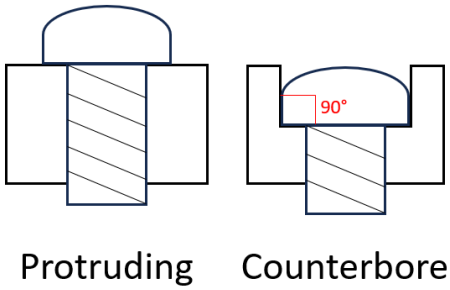
If the screws must be counterbore/recessed, any value given in millimeters (mm) (preferably greater than `--screw-size-top` or `--screw-size-bottom`) will create a circular shape to bury the head.

Default: 0 mm, protruding screw head.

Red: Screw head diameter

Note: Using recess for screws on bottom side requires manual flipping of the surface. `--separate-files` option may help you to achieve this.

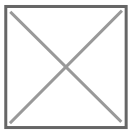
The images below show how `--screw-head-diameter` and `--screw-head-height` affect the creation of screw head related recesses.



`--screw-extra-margin`

By default, screws are added at the four corners of the top and/or bottom side of the box, leaving a margin of `--material-thickness` from the sides. If an additional margin is desired, any value given in millimeters (mm) will increase this gap.

Default: 0 mm, no additional margin.


Green: Screw extra margin

`--simulate`

If used, simulates the settings, without writing to actual files.

Default: false, write to actual files.

`--inkscape-path`

Path to Inkscape binary which is used when `--vectorize` is enabled (default).

Default: C:\Progra~1\Inkscape\bin\inkscape.com (on Windows), or Inkscape (on macOS), or inkscape (on Linux).

`--finger-joint`

Enables finger joints (box joints) on all 6 sides of the box so panels can be assembled without glue. By default, all 6 sides are enabled. To disable finger joints on specific sides, use the `--no-finger-<side>` arguments.

Default: false, no finger joints.

`--finger-size`

The width of each finger in millimeters (mm). If set to 0, the `--material-thickness` value will be used as the finger width.

Default: 0 mm (uses material thickness).

`--finger-count`

The number of fingers on an edge. If set to 0, the number of fingers will be calculated automatically based on the `--finger-size` value and the panel dimensions.

Default: 0 (automatic).

`--finger-clearance`

Extra clearance added to finger slots in millimeters (mm) to ensure smooth assembly.

Default: 0 mm.

`--no-finger-*`

Disable finger joints per side.

When `--finger-joint` is enabled, all 6 sides have finger joints by default. Use the following arguments to disable finger joints on specific sides:

- `--no-finger-top` — Disables finger joints on the Top panel
- `--no-finger-bottom` — Disables finger joints on the Bottom panel
- `--no-finger-front` — Disables finger joints on the Front panel
- `--no-finger-back` — Disables finger joints on the Back panel
- `--no-finger-left` — Disables finger joints on the Left panel
- `--no-finger-right` — Disables finger joints on the Right panel

`--help`

Shows the quick help with the list of arguments and their short descriptions.

JSON File

See the definition on this page: MIDILLI JSON File Definition.

Typical scenario for separate files

Below 5 different files are used to use 5 different settings on engraving:

1. Engrave text with a lower laser power and/or faster speed,
2. Cut components,
3. (If exists) Cut screw holes,
4. (If exists) Engrave component and/or screw recesses with a power slightly higher than text engraving and slower speed depending on engraving depth/screw head diameter.
5. Cut outlines to separate sides from the material

If you are using a laser engraving software similar to LaserGRBL (free and open-source), you can separately append the files one by one with different settings and have all engraving and cutting at once. Similarly, some other engraver software such as MeerK40t (free and open-source) let you also import all files at once and select the items to set different operation for each item.

Additional Resources

- Understanding Tool Diameter and Shape, Laser Kerf, and Nozzle Width in CNC, Laser, and 3D Printing
- JSON File Definition
- Understanding 2D Coordinate Systems in MIDILLI Tools