



# Post-processing scripts

**Relevant for**

PRUSASLICER

Last updated 6 months ago • [1 comment](#)

Article is also available in following languages



There are some things that PrusaSlicer simply doesn't do. However, using post-processing scripts you can automatically modify the generated G-code to do (almost) anything you want.

You can specify the path to the script in **Print settings - Output options - Post-processing scripts**.

The screenshot shows the PrusaSlicer software interface. The 'Print Settings' window is open, and the 'Output options' tab is selected. The 'Post-processing scripts' section is visible, showing a text field with the path: /Users/username/Documents/Shared/PrusaSlicer/Scripts/add\_counter.py; The interface also shows other settings like 'Sequential printing' and 'Output file'.



## Script setup

Post-processing scripts can be written in **any programming language** (Perl, Python, Ruby, Bash etc.). They just need to be recognized by your system as an executable and accept the path to the G-code file as the only argument.

If you want to run several scripts, put each script invocation into its own line.

## Script execution

Each script will be passed the absolute path of a **temporary** G-code file that PrusaSlicer generates. This file is stored in a temporary folder on your drive (typically your fast system drive). The script is then executed to modify the G-code in place and the resulting G-code file is written to your selected target folder or sent to a print host, such as PrusaConnect or Octoprint.

 Please note that the G-code viewer still visualizes G-code before post-processing.

PrusaSlicer automatically converts all environment variables used in post-processing scripts to uppercase. To ensure compatibility, always reference them using fully capitalized names, such as "SLIC3R\_FILL\_DENSITY".

Two additional environment variables are passed to post processing scripts: The environment variable SLIC3R\_PP\_HOST provides the host specification, where "File" means copying the G-code to a local hard drive or removable media, while the other values ("PrusaLink", "Repetier", "SL1Host", "OctoPrint", "FlashAir", "Duet", "AstroBox" ...) specify the print host type the G-code will be sent to.

The environment variable SLIC3R\_PP\_OUTPUT\_NAME contains the name of the G-code file including path (for SLIC3R\_PP\_HOST == "File") or a name that will be given to the file after it is uploaded to the host (PrusaLink, Octoprint ...)

The post-processing script may suggest a new output file name (likely based on SLIC3R\_PP\_OUTPUT\_NAME) by saving it as a single line into a new "output name" temp file, for example to add time stamps or sequence numbers to the final G-codes. The "output name" file name is to be created by suffixing the input G-code file name with ".output\_name". PrusaSlicer will read the new name of the file and handle it properly, for example, when sending it to Octoprint.

## Script Parameters

You can use parameters with your script as such:

`/path/to/executable` becomes `/path/to/executable` with the arg `outputfilename.gcode`

`/path/to/executable -arg -arg2` becomes `/path/to/executable` with args `-arg`, `-arg2`, and `outputfilename.gcode`

If path to executable or arguments contain spaces, these arguments need to be escaped using the escaping style common to the platform (shell escaping style on Linux and OSX, Windows command line escaping).

## Examples



### Adding counter to the file name example

Path to the final exported file (or its name for Octoprint)

```
env_slicer_pp_output_name = str(getenv('SLIC3R_PP_OUTPUT_NAME'))
```

Python script that modifies the output path

```
with open(sourcefile + '.output_name', mode='w', encoding='UTF-8') as fopen:  
    fopen.write(counter + '_' + env_slicer_pp_output_name)
```

## Historical changes and breaking backward compatibility

Before version 2.4, PrusaSlicer used to execute the script on the final G-code exported to your desired target medium. Which was very often a removable drive, such as an SD card. SD cards are slow and wear out, so this was not ideal. There were also problems when sending the G-code to Octoprint. The new approach with the temporary file on your system drive prevents these problems and extends the functionality of post-processing scripts. However, it might break some of your existing scripts.

This documentation page includes some texts from the [original Slic3r documentation](#) and from [Bob's Project Notebook](#).

---

### Was this article helpful?



[Log in](#) to post a comment

**aczekajski** • 4 months ago

TLDR:

Your program must read the gcode from file passed as last arg and write results to the same file.

Each line of the "Post-processing scripts" will be run as a terminal command with the gcode filepath added the end (as a last arg):

"node foo.js --bar" will be run as "node foo.js --bar /path/to/the/input.gcode".



[Reply](#)

## Blog

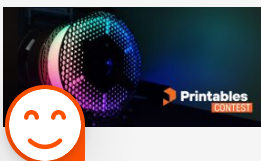
Prusa CORE One INDX – Unmatched Toolchanger With 8 Nozzles – Orders Now Open!



Remote 1080p/24fps Camera Streaming For Everyone and New Prusa Connect Storage Upgrade Options



Contest: Upcycling NEW Prusament Spools



## Newsletter

Signup and get our **monthly** squeeze on updates, blog, printables and much more.

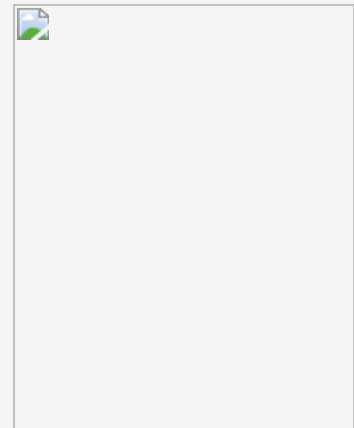
I would like to receive the e-book "Basics of 3D Printing" as a gift

By clicking, you agree to [receive our newsletter](#).


**Subscribe**

This site is protected by reCAPTCHA Enterprise and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

## Printables



**Tissue Paper Glider**





Introducing the Prusa Pro ACU: Why Overdrying is Bad for Your Filaments

**E-shop**

3D Printers

Printer Upgrades

Accessories

Prusa Merchandise



Filaments

**Printables**

3D Models

Store Clubs

3D Design

Contents

Brands

**Community**

World Map

Forum

User Groups

Events

**Help**

Prusa Help

Prusa Academy

3D Printing

Calculators

**Software**

Prusa Slicer

Prusa app

Prusa Connect

Drivers and Software

**Company**

About Us

Blog

Official

Prusa Resellers

Contact Us

Certified for

ISO 9001,

ISO 14001,

ISO 45001

General Terms and Conditions

General Terms and Conditions of Use of the PRUSA Websites

Privacy Policy



[Information about cookies](#)

[Customer Complaints Policy](#)

[Status Page](#)

---

[General Terms and Conditions](#)

| [General Terms and Conditions of Use of the PRUSA Websites](#)

| [Privacy Policy](#)

| [Information about cookies](#)

| [Customer Complaints Policy](#)

| [Status Page](#)

| [Do Not Sell or Share My Personal Information](#)

| [Supplemental Statements](#)



JOSEF PRUSA®, PRUSA RESEARCH®, PRUSA POLYMERS®, PRUSA ORANGE®, PRUSA 3D® AND PRUSAMENT® are registered trademarks of Prusa Development a.s. used by Prusa Research a.s. under licence from Prusa Development a.s.

© Prusa Research a.s.

