

KY-040 Filament Motion Sensor

L LeFish

VIEW IN BROWSER

updated 7. 4. 2023 | published 7. 4. 2023

Summary

A Filament Motion Sensor based on a KY-040 rotary encoder. Sensor is working stable - long term stability to be tested.

[3D Printers](#) > [3D Printers - Upgrades](#)

Tags: [filamentsensor](#)

Credits first:

The credit for the filament path go to Jason (https://www.printables.com/de/@Jason_116929)

I hacked this together with parts at hand in my scrap box. Feel free to mod and optimize!

The picture above shows the sensor after around 100m of filament of the poorest quality you can think of. This very filament was the reason for me to hack together this sensor.

Hardware part:

The following parts are needed for this version additionally to the printed parts to mount this on a Voron 2.4:

- 12.7mm dremel sanding drum i.e. shorturl.at/bHP69
- rotary encoder KY-040 i.e. shorturl.at/bfvQ2
- 2 bearings 8x16x4mm
- 2 cables for the encoder
- PTFE-Tube for filament lining through the sensor (length around 50 mm)
- 3 self-tapping screws 3.5x20mm
- 1 SHCS M3x12mm (→Voron)
- 1 M3 T-Nut (→Voron)

Line up the drum and the adapter and the KY-040. It is supposed to form a tight fit. Push the bearings over the adapter. Put the whole assembly into the bottom housing.

Push the PTFE tube lining into the top housing. Cut away the piece of the PTFE tube that shows in the middle of the sensor to reveal the filament in this area for the drum to touch it.

Solder the two cables to any of the encoder pins. Test with a multimeter if you get a signal when manually rotating the drum.

Assemble top and bottom part and tighten them with the self-tapping screws.

Repeat the same test with a multimeter when manually sliding a filament through the sensor. If you do not get a stable signal you can tighten the drum screw to enlarge its diameter.

PS: If you have the more common 8x16x5mm at hand simply widen the bearing cutouts in the housing parts (STP file of assembly included).

Software part (klipper):

I added the filament sensor the following way:

printer.cfg:

```
[filament_motion_sensor FilamentMotionSensor] detection_length: 4.0 #  
The minimum length of filament pulled through the sensor to trigger # a  
state change on the switch_pin # Default is 7 mm. extruder: extruder #  
The name of the extruder section this sensor is associated with. # This  
parameter must be provided. switch_pin: ^!PB14 # Pin (Endstop-Pin auf  
SPider Board) pause_on_runout: false runout_gcode: FILAMENT_ISRUNOUT
```

```
#insert_gcode: #event_delay: 3.0 #pause_delay: 5 # See the
"filament_switch_sensor" section for a description of the # above
parameters. [gcode_macro M600] gcode: {% set X = params.X|
default(50)|float %} ;{% set Y = params.Y|default(0)|float %} ;{% set Z =
params.Z|default(50)|float %} SAVE_GCODE_STATE NAME=M600_state
CLEAR_PAUSE PAUSE G91 G1 E-.8 F2700 G1 Z50 F600 G90 G1 X10 Y10
F10000 G92 G92 E0 ;G1 E-50 F1000 ;G1 E-30 F1000
RESTORE_GCODE_STATE NAME=M600_state [delayed_gcode
RECHECK_FILAMENT_ISRUNOUT] gcode: {% set runout_state = "deadtime"
if printer.print_stats.print_duration < 30 else "runout" %} {% set
runout_state = "not_printing" if printer.idle_timeout.state != "Printing" %}
{% if runout_state == "deadtime" %} ## During initial print dead time
sensor value is not valid. { action_respond_info("No runout detection
below 30 s print duration time.") } {% elif runout_state == "not_printing"
%} ## Printer not in printing state { action_respond_info("No runout
detection as printer is not in printing state.") } {% else %} {
action_respond_info("Detected actual runout. Go to filament change
position.") } M600 ; Go To Filament Change position {% endif %}
[gcode_macro FILAMENT_ISRUNOUT] gcode: UPDATE_DELAYED_GCODE
ID=RECHECK_FILAMENT_ISRUNOUT DURATION=5 ; Recheck after XXs
duration { action_respond_info("Detected possible runout. Recheck runout
sensor in 5s.") }
```

In **main sail.cfg** add

```
[delayed_gcode SETFILAMENTSENSOR_ON]
gcode:
SET_FILAMENT_SENSOR SENSOR=FilamentMotionSensor ENABLE=1
```

and

```
UPDATE_DELAYED_GCODE ID=SETFILAMENTSENSOR_ON DURATION=5
```

at the very end of [gcode_macro RESUME]

Also add

```
SET_FILAMENT_SENSOR SENSOR=FilamentMotionSensor ENABLE=0
```

at the very end of [gcode_macro PAUSE]

This implementation ensures that the sensor values are overwritten by “filament present” during a dead time of 30 seconds after print start of resume.

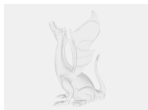
The sensor is disabled during pause state to freely work with extruder to fix the filament issue.

Happily print filament rather than air!

Model files

filament_motionsensor-adapter.stl

filament_motionsensor-gehause_unten.stl



filament_motionsensor.step

filament_motionsensor-gehause_oben.stl

License ©



This work is licensed under a
Creative Commons (4.0 International License)

Attribution—Noncommercial—Share Alike

- ✗ | Sharing without ATTRIBUTION
- ✓ | Remix Culture allowed
- ✗ | Commercial Use
- ✗ | Free Cultural Works
- ✗ | Meets Open Definition