BD Sensor

- https://github.com/markniu/Bed_Distance_sensor/wiki

YouTube Video #108



Übersicht

- https://en.wikipedia.org/wiki/Eddy_current
- 2 Versionen
  - BDsensor → Sensor & Elektronik im Bauraum
  - BDsensorM → nur Probe im Bauraum und Elektronik steckbar auf EXT1 Port
- kein eigenes USB / CAN Interface !
- Nutzt I2C für die Kommunikation
  - Der Sensor funktioniert mit einem Software I2C Port
- Es muss eine Erweiterung für Klipper installiert werden!

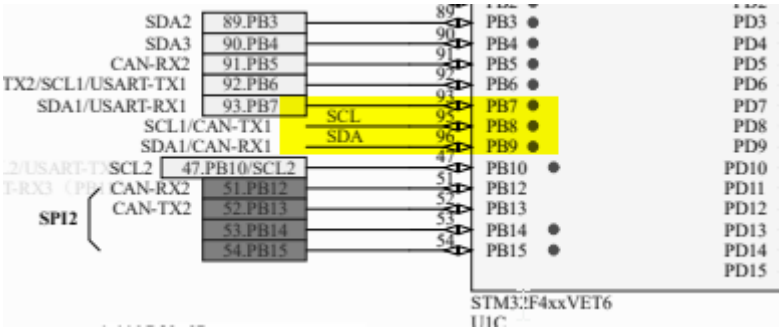
Anschluss am Board

- Der Sensor muss an einem bestehenden Druckerboard per I2C angeschlossen werden.
- Es werden zwei Pins für I2C benötigt: SDA und CLK und zwei für **5V** und GND
- I2C muss nicht zwingend ein Hardware I2C sein. Die sind oft eh schwer auszumachen. Es reichen 2 Pins auf denen dann ein Software basierter I2C läuft. Die Umsetzung macht Klipper lleine 🙂
- Wenn der Sensor Real time Leveling machen soll muss er an dem Board angeschlossen sein, an dem auch die Z motoren angeschlossen sind. Siehe https://github.com/markniu/Bed_Distance_sensor/wiki/How-to-run-Real-Time-Level
- Als Pins kann man u.A. am kopfboard einen BLTouch Anschluss verwenden. Der hat in der Regel auch gleich 5V und GND. An einem normalen Druckerboard kann man meist einen EXT1/ETX2 Anschluss verwenden. Generell müssen es Pins sein, die nicht schon durch irgendwelche Logik oder Filter beschaltet sind - also reine IO Pins!

Beispiel für den Anschluss am BLTouch Port:

```
BLtouch    |    BDsensor
 GND       -->     GND
 5V        -->     5V
 S         -->     CLK/SCL    (Input)
 GND       -->     GND
 Zmin      -->     SDA    (Input/Output)
```

Beispiel für den Anschluss am Spider V1.0:



Klipper kompilieren

- die BLSensor Repo klonen
  - `cd ~`
  - `git clone https://github.com/markniu/Bed_Distance_sensor.git`
- BLSensor Addons installieren
  `~/Bed_Distance_sensor/klipper/install_BDsensor.sh`
- Klipper kompilieren
  - `cd ~/klipper/`
  - `make menuconfig`
    Hinweis Hier müsst ihr euer Board ganz normal einstellen. Es gibt keine direkte Option für BDSensor!
  - `./make_with_bdsensor.sh`
  - `make flash`
    Hinweis Wie ihr genau euer Board flashen müsst hängt vom Board und der Flashmethode ab! `make flash` steht hier nur als Beispiel und ihr müsst das zwingend an euer Board bzw. eure Verbindung anpassen!
  - Darauf achten das während dem Flashen folgendes erscheint : `Compiling out/src/BD_sensor.o` während man `make_with_bdsensor.sh` ausführt
- Updates kann man auch machen - bedarf aber dann immer einem Klipper Update auf dem Board! eine kurze Beschreibung dazu findet ihr hier: https://github.com/markniu/Bed_Distance_sensor/wiki/Installing-for-Klipper

erster Test

- sicherstellen das `[probe]` und oder `[bltouch]` in eurer Konfig nicht mehr aktiv sind
- am besten eine eigene Konfig Datei anlegen für den Sensor → `BDsensor.cfg`
- für einen ersten Test reicht dann folgender Inhalt:

```
[BDsensor]
# Don't use aliases for the board pins
```

```
sda_pin: PB1 # example of connecting to main board Creality V4.2.7
scl_pin: PB0
#scl_pin:MKS_THR:gpio20 # example of connecting to CAN module like MKS
THR42
#sda_pin:MKS_THR:gpio11
#scl_pin:host:gpio17 # example of connecting to GPIO on RaspberryPi
#sda_pin:host:gpio27
delay: 20 # you can set it 10 if the BDsensor version is >=1.2
z_offset:0 #  within -0.6 to 0.6mm
x_offset: -34
y_offset: 0
no_stop_probe: # fast probe that the toolhead will not stop at the
probe point,disable it by commenting out.
position_endstop: 1.2 #the triggered position, recommend value is 1~2.8
collision_homing:0 #  set it 1 to enable homing with nozzle collision
sensing.
collision_calibrate:0 # set it 1 to enable auto calibrate BDsensor with
nozzle collision sensing.
#QGL_Tilt_Probe:0 #set 1 to enable probe up and down when do
quad_gantry_level
```

WICHTIG Ihr müsst hier die I2C Pins korrekt eintragen! Also sda_pin und scl_pin anpassen. Das hängt natürlich von eurem Board ab.

- Wenn ihr jetzt in der MainSail Konsole `M102 S-1` ausführt müsste folgendes Ergebnis kommen:

```
BDsensorVer:V1.2c pandapi3d
,switch_mode=1,collision_homing=0,collision_cal=0
```

- Wenn der Sensor nicht korrekt angeschlossen oder klipper falsch kompiliert wurde, dann bekommt ihr folgendes:

```
No data or corrupt data from BDsensor(□□□□□□□□□□□□□□□□□□□□□□), Please
check connection
```

Einbau

Konfiguration

BD Sensor M → Falsche Pin Config !
Pins sind 2 Pins verschoben für Clock und SDA

- [BDsensor] Sektion
- Make sure there are [safe_z_home],[bed_mesh], [force_move] in your printer.cfg .
- [stepper_z] → H7
    - endstop_pin : probe:z_virtual_endstop
    - homing_speed : 8
    - second_homing_speed : 1.4 #set this to 3 if homing with collision

    - homing_retract_speed : 2
    - homing_retract_dist : 0
- [bed_mesh]
    - horizontal_move_z : 2
    - probe_count : 9,9
    - zero_reference_position : 150, 160 # Set this value to be the same as home_xy_position that is in the section safe_z_home

Bed Mesh

Nozzle Collision Sensing

- https://github.com/markniu/Bed_Distance_sensor/wiki/Collision-sensing
- WICHTIG → Retract auf 0 !!!!! https://github.com/markniu/Bed_Distance_sensor/wiki/Homing-with-contactless-and-contact-probing-together

Firmware BD-Sensor

Die Firmware vom BDSensor kann man bei Bedarf neu Flashen. Dazu gibt es auf der Wiki Seite passende Anleitungen:

- Für Hardware Version 1.1 https://github.com/markniu/Bed_Distance_sensor/wiki/Flash-BDsensor(V1.1)-firmware
- Für Hardware Version >= V1.2 https://github.com/markniu/Bed_Distance_sensor/wiki/Flash-BDsensor(stm32)-firmware

Hardware

- X Carriage Voron Default https://github.com/VoronDesign/Voron-2/tree/Voron2.4/STLs/Gantry/X_Axis/X_Carriage
- x Carriage Voron mit X Endstop https://mods.vorondesign.com/details/snBnPF4Mo9nfFMZdahC9Ng
- Voron Trident officially uses a slightly improved version of this mod. It is suggested to use the X axis parts from Trident instead https://github.com/VoronDesign/Voron-Trident/tree/main/STLs/Gantry/X_Axis
- Lazy BD-SENSOR adapter For Stealthburner Voron https://www.printables.com/model/831679-lazy-bd-sensor-adapter-for-stealthburner-voron

Links

- https://www.pandapi3d.com/bdsensor
- https://github.com/markniu/Bed_Distance_sensor
- Klipper Doku - Eddy Current Inductive probe

https://www.klipper3d.org/Eddy_Probe.html

ERROR Unknown command: I2CBD

```
MCU Protocol error

This is frequently caused by running an older version of the
firmware on the MCU(s). Fix by recompiling and flashing the
firmware.

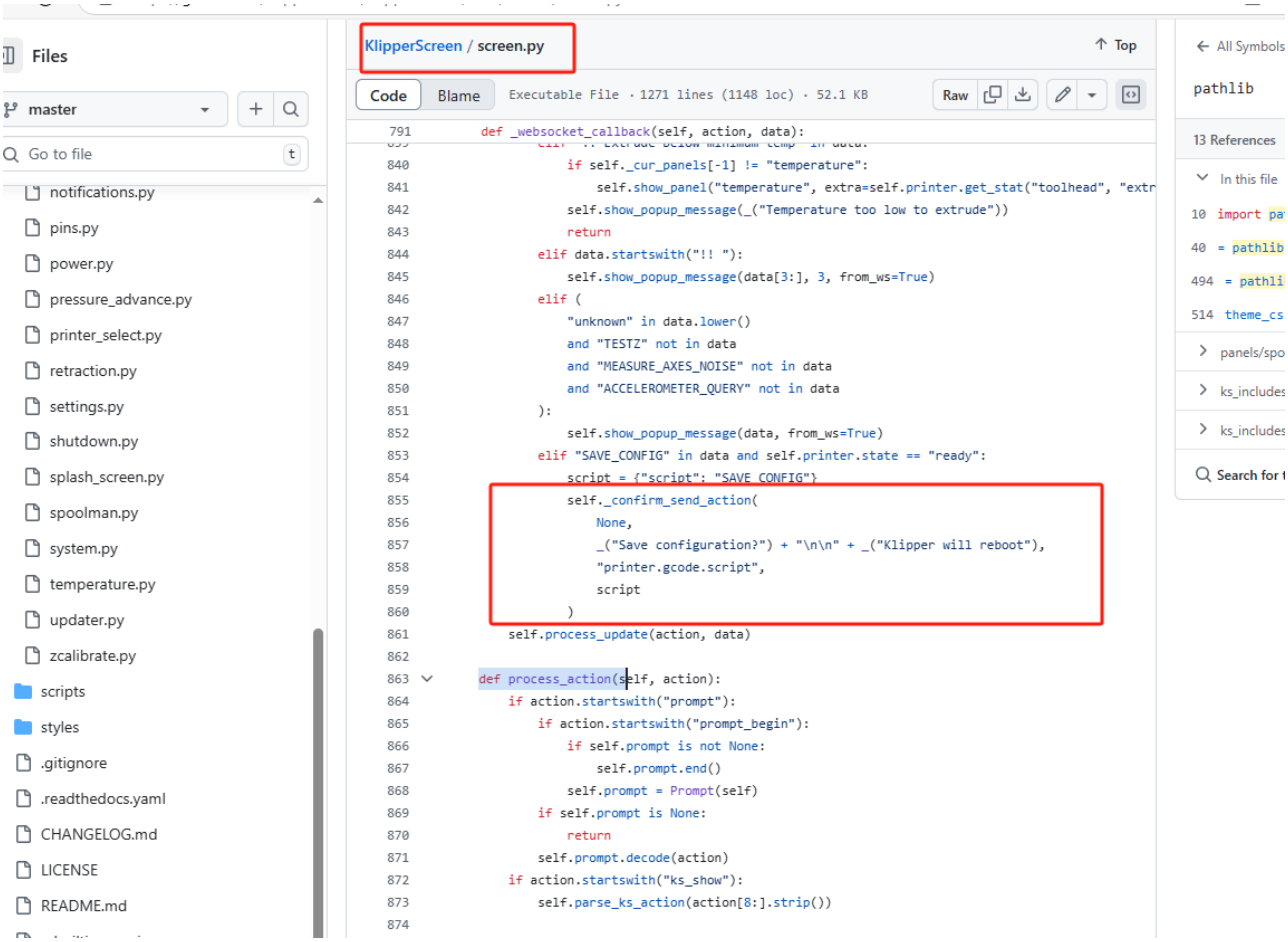
Your Klipper version is: v0.12.0-349-ga34034494-dirty
MCU(s) which should be updated:
mcu: Current version v0.12.0-349-ga34034494
RasPi: Current version v0.12.0-349-ga34034494
EBBCan: Current version v0.12.0-349-ga34034494
Up-to-date MCU(s):
<none>

Once the underlying issue is corrected, use the "RESTART"
command to reload the config and restart the host software.

mcu 'mcu': Unknown command: I2CBD
```

NOTES

- Kein Collision aktiv vor Kalibrieren ! Kopf macht sonst bei M102 S-6 ein homing und fährt u.U- ins Bett !
- Collision Sensing
  - Raw data:467 at 0 mm, BDsensor to bed: 1.8680 mm, Bed: 42.2 → Kopf zu nah
  - Homing Speed max auf 2 eher 1 → Sonst setzt die Nozzle auf !
  - warning: triggered in air, 741
- horizontal_move_z : 10
  - max 1-2
  - Messbereich Spule Bed → max 3,5mm
- Bed Mesh Save Meldung auf Screen



Startmakro mit 150 Grad !!! → Fila vorweichen

Nozzle Clean → Umbauen !

From:
https://drklipper.de/ - **Dr. Klipper Wiki**

Permanent link:
https://drklipper.de/doku.php?id=klipper_faq:eddy:bd_sensor&rev=1736692309

Last update: **2025/01/12 15:31**