BTT Eddy USB setup

Principles

1. The BTT **Eddy measures frequency** between the conductive part of the build plate and its coil to calculate distance. Therefore the **Eddy does not and cannot rely on Z-Offset**. It only truly has X and Y offsets from your nozzle.
2. The **frequency measured is affected by the heat of components inside the eddy**. Thus calibrating thermal compensation is essential.
3. In order for the eddy to know how high off the bed it is, initially you need to tell it … "Hey Eddy! I've set the nozzle to Z=0…. Now test frequencies at that spot and map those frequencies to Z=0." - a.k.a. **the paper test**.
4. The **homing XY point** and the **eddy probe calibration point** should be the same point!
5. Choose a **homing XY** point that has the least amount of physical variance when heating the bed. I have found the most consistent point is my reference bed screw… in my case at the rear left of the bed. If you don't do this, you will run into major z issues after thermal calibration.
6. The process of **mapping frequencies for z-height** at different temperatures (of the internals of the Eddy; not bed or the nozzle temps) **is a manual process** and will likely take an hour or more of concentration.

Setup Overview

1. Design and print an Eddy Mount if required for your setup.
2. Connect Eddy.
3. Update and Patch the main Klipper to support Eddy.
4. Configure Klipper (printer.cfg) for Eddy support.
5. Setup initial Z - "eddy paper-test"
6. Do thermal compensation calibration
7. Set the true Z - by printing and measuring and adjusting the probe calibration (no paper)
8. CAUTION

Setup Details

1. Design and print an Eddy Mount if required for your setup

- Just do it … I made my own for my Micro Swiss NG extruder

2. Connect Eddy

- Connect to a well-powered USB port - I'm using an externally powered

USB hub.

3. Update and Patch the main Klipper to support Eddy

- Perform the normal Mainsail/Fluid updates to make sure you're on a current version of Klipper > v205
- Download the "eddy.patch" file to your klipper host:
  - I put this in my klipper user's ~/patch directory.
- Install the patch:
  - ~/Klipper/patch -p1 < ~/patch/eddy.patch
- Restart Klipper
- Flash the Eddy and any MCU boards.

4. Configure Klipper (printer.cfg) for Eddy support.

- In "printer.cfg" :
  - Remove your existing bltouch or similar config
  - Add the following:

```
[mcu eddy]
serial: /dev/serial/by-id/usb-Klipper_rp2040_45474E621A8BE41A-if00 # EDDY
<— replace this with your actual eddy serial device

[temperature_sensor btt_eddy_mcu]
sensor_type: temperature_mcu
sensor_mcu: eddy

[probe_eddy_current btt_eddy]
sensor_type: ldc1612
z_offset: 1.0 # Set to a non-zero value
i2c_mcu: eddy
i2c_bus: i2c0f
x_offset: -40.949    # Set actual offset relative to nozzle
y_offset: -10.317     # Set actual offset relative to nozzle
data_rate: 500

[temperature_probe btt_eddy]
sensor_type: Generic 3950
sensor_pin: eddy:gpio26
horizontal_move_z: 2
```

  - In the [safe_z_home] section update your homing position to be stable point for homing at different temperatures

```
[safe_z_home]
home_xy_position: 74, 204 # these are my home XY coordinations above rear-
left bed screw… set the values for your homing location
```

    - (Optional) Set your bed_mesh speed to 100 if you’re happy with that.

```
[bed_mesh]
speed: 100
```

    - Add macros:

```
; ------------------------------------ EDDY PROBE  MACROS ---------------------------------------------------------------------


[gcode_macro nozzle_home_XY]
description: moves the nozzle xy to reference bedscrew location
gcode:
  G0 X33.05 Y193.68 F6000 # these are my home XY coordinations above rear-left bed screw… set the values for your homing location


[gcode_macro paper_test]
description: paper_test (assumes z is homed and z_tilt already set)
gcode:
  BED_MESH_CLEAR
  G28 X Y
  NOZZLE_HOME_XY
  G0 Z10
  PROBE_EDDY_CURRENT_CALIBRATE CHIP=btt_eddy


[gcode_macro eddy_mesh]
description: perform a bedmesh .. optional parameter “PROFILE” will default to “default”
gcode:
  {% set profile = params.PROFILE|default("default") %}
  BED_MESH_CALIBRATE METHOD=scan SCAN_MODE=rapid PROFILE={profile}
```

    - Do a “FIRMWARE_RESTART”

5. Set initial Z - “eddy paper-test”

- With a cold bed:
    - G28
    - PAPER_TEST
- Get the nozzle to very lightly grip the paper
    - (be very aware of how much tension there is on the paper - and keep this consistent for all future paper tests)

- Lower Z by another 0.12mm (the paper thickness)
- Accept
- Now the eddy moves over the probe point and does some bounciness to calibrate z-frequencies for that z-height
- SAVE_CONFIG

6. Do thermal compensation calibration

This is a very manual process. We need to heat-soak the Eddy i.e. get the Eddy internal temperature as hot as it will go when the printer is at your highest-temperature printing environment. In my case I print ABS with a bed temp of 90C, with the nozzle anywhere between 250C - 280C.

- Start cold ….. and note what your “BTT Eddy” temperature is at e.g. 35C … note you don’t care about the “BTT Eddy Mcu” temperature, just the “BTT Eddy” temperature.
- Start your bed heating to 100C or whatever the highest bed temp is
- Start heating your nozzle to your highest printing temp e.g. 280C in my case.
- Move the Eddy to the centre of the bed and set your nozzle Z height to around 1mm (we want that Eddy surrounded by heating pain!)
- Wait until the “BTT Eddy” temp stabilizes - this will take some time. Mine got to around >70C, with my bed at 100C and enclosure closed. Make a note of this temperature as your **Max Eddy Temperature**
- Now .. the painful part:
    - Cool down your machine until the “BTT Eddy” gets back to the cold temp you noted in the first step e.g. 35C or close enough to that.
    - Once cool, you can start the thermal compensation test:

NOTE: for this test I created some bash scripts as follows to be able to control things from the command line during the test:

```
biqu@BTT-CB1:~/bin$ cat bed_heat
echo "SET_HEATER_TEMPERATURE HEATER=heater_bed TARGET=$1" >~/printer_data/comms/klippy.serial
```

```
biqu@BTT-CB1:~/bin$ cat nozzle_papertest
echo "G0 X33.05 Y193.68 F6000" >~/printer_data/comms/klippy.serial # this is my rear-left bed screw location for the nozzle
```

```
biqu@BTT-CB1:~/bin$ cat nozzle_center
echo "G1 X135 Y135 Z1 F6000" >~/printer_data/comms/klippy.serial # this is roughly the center of my bed
```

-
    - PROBE_DRIFT_CALIBRATE PROBE=btt_eddy TARGET=70 STEP=5

    - The TARGET in the above is your **Max Eddy Temperature** noted before. What this test does, is wait for you to manually get the temperature *of the BTT Eddy* to go up slowly and then you measure z=0 with a paper test at each 5C increment of eddy temp, and it calibrates. The important thing here is to try and keep the “eddy temp” at a consistent temperature while it’s calibrating! As you raise the bed temp, as soon as the eddy heat crosses its next 5C step, it will prompt you for a paper test. As soon as it does this, you may want to lower the bed temp by 2C or so to avoid too much overshoot while paper-testing and calibrating.
    - As soon as the test starts it will want you to do a paper test at your probe location (my rear-left bed screw)
        - Use the Nozzle_papertest bash script on your command line to move your nozzle to your probe location
        - Do the papertest here and accept, it will then probe that location to map the frequencies at the current “eddy temp” - again, try to keep the “eddy temp” stable for this operation. - I use the “bed_heat” bash script for this purpose to tweak the bed temp to keep the “eddy temp” consistent.
    - Once it’s completed, move the eddy back to the centre of the bed with the “nozzle_center” bash script, so that it can heat soak again.
    - Raise the temperature of the bed in 5C increments until the “btt eddy” temp increases to its next 5C threshold….
    - Move the nozzle back with “nozzle_papertest” script… adjust bed temp to stabilize “Btt Eddy” temp, and do the paper test again ….
    - You’re probably seeing the cycle now …. Keep looping through (raising eddy temp, paper test), until you get to your Max Eddy Temperature threshold… then the test will complete
- SAVE_CONFIG
    - This will save all the mapped frequencies to the temperatures into the bottom of your printer.cfg

7. Set the true Z - by printing and measuring and adjusting the probe calibration (no paper)

- Print a simple single layer (0.25mm) square with a 5 loop skirt
- Measure the height of the skirt with some precise calipers. If the height is for example 0.29mm then the z is too high by 0.04mm. So we need to subtract that from the configured z=0.
- With a stable temp bed (cold or any other consistent temperature):
    - G28

    - PAPER_TEST
- *** DO NOT USE PAPER ***
- **In this example** we're 0.04mm too high, so:
    - Our starting offset is 10mm
    - Lower the nozzle by 0.04mm
    - Then lower in 1mm increments until we're at -0.04mm
- Accept
- Now the eddy moves over the probe point and does some bounciness to calibrate z-frequencies for that z-height
- SAVE_CONFIG

8. CAUTION

- BEWARE : Using different build plates or double-sided build plates
    - Due to the physical layers on your build plate, there is often variance in the height of the PEI layer / textured PEI layer from one build plate to the next. These layers are non-conductive material which sit above the conductive steel. The eddy will ignore non-conductive material, so this physical difference will affect your build height. In this case, after all of your configuration on one plate (or side of a plate), you will have a reference plate for Z=0, which is now hard-stored in your printer.cfg file. To adjust the z height from one plate to the next, you will need to print and measure a skirt height to find out the variance in mm of your build plate surface. In my case there is a 0.2mm difference between my PEO surface and my textured PEI surface. Once you have that number, you can configure plate profiles in ORCA slicer, and detect the plate type in your start gcode in order to affect the z-offset accordingly : see Orca's "working with multiple bed types" page here https://github.com/SoftFever/OrcaSlicer/wiki/bed-types#multiple-bed-types
- BEWARE : Never again will you use bed-mesh from mainsail UI
    - From now on you will use the EDDY_MESH macro we added above. If you are using adaptive mesh, be sure to add the parameters: METHOD=scan SCAN_MODE=rapid
- BEWARE : you cannot calibrate and save z height by using z-offset.
    - Once Z=0 is truly set... making changes to the z-offset and saving will have no effect. To change the Z-height you must use Probe_eddy_current_calibrate (see PAPER_TEST macro above).
    - You can still micro step during a print or in gcode, but z-offset changes won't save. After the z-motors are disengaged, eddy uses frequency to discover z=0, not the z-offset.
- BEWARE: UPDATING KLIPPER in the future..... because we're using a patched version of Klipper your updates will need to happen like this, until the code is officially merged into klipper main branch:

- git stash
- do mainsail update stuff
- git stash pop
- klipper restart
- klipperscreen restart