Включение отображения температуры Host, Head MCU, Back MCU, MCU Box

v.2 UPD 22/10/25

- Уточнение названия чипов
- Для MCU головы, изменены коэффициенты. Взяты из документации на под серию GD32F303xB вместо более общей документации GDF303xx, как в первой версии. На выходе получаем, что MCU головы греется сильнее на 8-10 градусов. Спасибо коллегам за тесты с тепловизором и термопарой.

Внимание! Датчики внутри MCU с завода не калибруются и различные экземпляры могут выдавать разные погрешности. Данные температуры не являются точными и используются только для общей оценки состояния системы.

Данное руководство подразумевает, что вы владеете навыками работы с конфиг файлами, ssh, sftp и умеете редактировать/копировать файлы в файловой системе принтера.

Все изменения вы делаете на свой страх и риск, и под личную ответственность!!!

Как известно Klipper создавался под такие микроконтроллеры как STM32, однако китайцы не стоят на месте и давно научились делать их функциональных клонов. Благодаря, тому что разработчики принтера переопределили и жестко указали тип MCU, Klipper даже и не знает, что работает с клонами.

Итак в Qidi Q2 установлены:

- Host: Rockchip RK3308B, поддерживается на уровне ядра linux, Отображение температуры **работает** в стоке, проблем нет.
- MCU Head: GD32F303CBT6 (переопределен как stm32f103) , используются коэффициенты от STM32, которые могут искажать настоящую температуру. Отображение температуры р**аботает** в стоке, но **желательна*** правка коэффициентов от клона.
- MCU Back: GD32F425RET6 (переопределен как stm32f407). У оригинального STM32 калибровочные коэффициенты при производстве записываются в спец регистры,

которых у китайского клона нет. Отображение температуры **не работает** в стоке, при активации Klipper падает в ошибку (Division by zero). **Требуется*** изменение расчета температуры и использование коэффициентов от клона.

1. Нужно заменить файл *home/mks/klipper/klippy/extras/temperature_mcu.py* на приложенный в архиве. [ссылка на архив]

Обязательно сделайте копию оригинального файла.

Вы можете не заменять файл на предложенный, а отредактировать ваш:

Нужно изменить два метода в данном файле:

```
def config_stm32f1(self):
        self.slope = 3.3 / -.004070
        self.base_temperature = self.calc_base(25., 1.454 / 3.3)
```

и

```
def config_stm32f4(self):
        self.slope = 3.3 / -.004400
        self.base_temperature = self.calc_base(25., 1.4 / 3.3)
```

****//

Все коэффициенты взяты из официальных даташитов GD32.

2) В printer.cfg добавьте температурные датчики:

Хост:

```
[temperature_sensor RK3308B]
sensor_type: temperature_host
min_temp:0
```

```
max_temp:100
```

MCU в голове:

```
[temperature_sensor MCU_Head]
sensor_type: temperature_mcu
sensor_mcu: THR
min_temp:0
max_temp:100
```

MCU на задней плате (!Только если вы выполнили пункт 1, иначе получите ошибку Klipper!):

```
[temperature_sensor MCU_Back]
sensor_type: temperature_mcu
sensor_mcu: mcu
min_temp:0
max_temp:100
```

После всех правок перезагружаем принтер с кнопки питания сзади

Включаем в настройках fluidd:

image.png

И получаем:

%D0%BE%D1%82%D0%BE%D0%B1%D1%80%D0%B0%D0%B6%D0%B5%D0%BD%D0%B8%

P.S. После обновления прошивки будет необходимо проделать все операции заново!

Автор: @EteRys

Включение температур плат BOX

Я не знаю на кой чёрт я это сделал.
Вывод температур mcu box для тех кто вывел температуры прочих mcu. Работает на 1.1.0, должно работать на 1.1.1.

temperature_mcu.py. берем готовый, либо заменяем в нем код:

```
1. Заменить метод config_stm32f4

Было (старый вариант с выводом mcu back):

def config_stm32f4(self):
    self.slope = 3.3 / -.004400
    self.base_temperature = self.calc_base(25., 1.4 / 3.3)

Стало:

def config_stm32f4_dispatch(self):
    mcu_name = self.mcu_adc.get_mcu().get_name()
    if mcu_name in ('mcu_box1', 'mcu_box2', 'mcu_box3', 'mcu_box4'):
        cal_adc_30 = self.read16(0x1FFF7A2C) / 4095.
        cal_adc_110 = self.read16(0x1FFF7A2E) / 4095.
        self.slope = (110. - 30.) / (cal_adc_110 - cal_adc_30)
        self.base_temperature = self.calc_base(30., cal_adc_30)
    else:
        self.slope = 3.3 / -.004400
        self.base_temperature = self.calc_base(25., 1.4 / 3.3)

2. В cfg_funcs заменить строку

Было:

('stm32f4', self.config_stm32f4),

Стало:

('stm32f4', self.config_stm32f4_dispatch),
```

3. Проверить config_stm32f0x2

Он остался в виде вызова self.config_stm32f4(...) — заменить на самостоятельную реализацию, иначе будет AttributeError (метода config_stm32f4 больше нет):

Было:

```
def config_stm32f0x2(self):
    self.config_stm32f4(addr1=0x1FFFF7B8, addr2=0x1FFFF7C2)
```

Стало:

```
def config_stm32f0x2(self):
    cal_adc_30 = self.read16(0x1FFFF7B8) / 4095.
    cal_adc_110 = self.read16(0x1FFFF7C2) / 4095.
    self.slope = (110. - 30.) / (cal_adc_110 - cal_adc_30)
    self.base_temperature = self.calc_base(30., cal_adc_30)
```

4. Отступы и проверка

    def config_stm32f4_dispatch(self): — 4 пробела;

    тело функции — 8 пробелов;

    тело if/else — 12 пробелов.

5. В файл boxN.cfg (где N = 1…4)

Добавить один раздел — датчик температуры MCU этого бокса:

```
[temperature_sensor MCU_Box1]
sensor_type: temperature_mcu
sensor_mcu: mcu_box1
min_temp: 0
max_temp: 100
```

Для остальных боксов — аналогично, с заменой цифры:

    бокс 2 → [temperature_sensor MCU_Box2], sensor_mcu: mcu_box2

```
бокс 3 → [temperature_sensor MCU_Box3], sensor_mcu: mcu_box3

бокс 4 → [temperature_sensor MCU_Box4], sensor_mcu: mcu_box4
```

Автор: Кирилл

Revision #9
Created 2026-10-08 09:40:30 UTC by Volvik88
Updated 2026-11-16 13:44:17 UTC by Volvik88