SKR Mini E3 1.2 Klipper Config/Guide (including stock LCD)

Hello all,

I was using klipper before i got the e3 mini, so i wanted to go back to it on my new board.

It took a bit of fiddling to get it to work, especially the LCD, and i could not find any comprehensive guides, so i want to share the installation process and the klipper host config.

First, when you are installing the board make sure to swap case/electronics fan with the part-cooling fan. This not really documented anywhere, so i want to drop this as a side note. Otherwise your fan speed settings will control the case fan and part cooling fan will be 100% on all the time.

I am not going to explain how to install octopi or download klipper from scratch, you can follow the guides on their respective websites.

When you are installing klipper, there is a `make menuconfig` step. This step let's you enter the options for the firmware that you will flash to the board, and you need to enter correct board settings:

Micro-controller Architecture	STMicroelectronics STM32
Processor model	STM32F103
Bootloader offset	28KiB bootloader

Then check(with spacebar) `Enable extra low-level configuration options` at the top and enter `!PC13` to `GPIO pins to set at micro-controller startup`.

You will basically end up with this:

https://i.imgur.com/Tw7ObWE.png

Exit and keep following the guide. After `make` step there is a `make flash` step but this does not work with this board.

At this point you should have a `klipper.bin` file in `out` directory(~/klipper/out/kipper.bin), get this file from your raspberry pi (use ftp or scp or plug the sd card on your computer to get the file, you can use winScp on windows).

Rename `klipper.bin` to `firmware.bin` and put it your printer's sdcard. When this file is present printer flashes itself with that firmware.

After that there are steps for configuring klipper host by creating a `printer.cfg` file.

Here is the contents of the file:

```
[stepper_x]
step_pin: PB13
dir_pin: !PB12
enable_pin: !PB14
step_distance: .0125
endstop_pin: ^PC0
position_endstop: 0
position_max: 235
homing_speed: 50

[tmc2209 stepper_x]
uart_pin: PB15
microsteps: 16
run_current: 0.580
hold_current: 0.500
stealthchop_threshold: 250

[stepper_y]
step_pin: PB10
dir_pin: !PB2
enable_pin: !PB11
step_distance: .0125
endstop_pin: ^PC1
position_endstop: 0
position_max: 235
homing_speed: 50

[tmc2209 stepper_y]
uart_pin: PC6
microsteps: 16
run_current: 0.580
hold_current: 0.500
stealthchop_threshold: 250

[stepper_z]
step_pin: PB0
dir_pin: PC5
```

```
    enable_pin: !PB1
    step_distance: .0025
    endstop_pin: ^PC2
    position_endstop: 0.0
    position_max: 250

    [tmc2209 stepper_z]
    uart_pin: PC10
    microsteps: 16
    run_current: 0.580
    hold_current: 0.500
    stealthchop_threshold: 5

    [extruder]
    step_pin: PB3
    dir_pin: !PB4
    enable_pin: !PD2
    step_distance: 0.010526
    nozzle_diameter: 0.400
    filament_diameter: 1.750
    heater_pin: PC8
    sensor_type: EPCOS 100K B57560G104F
    sensor_pin: PA0
    control: pid
    pid_Kp: 21.527
    pid_Ki: 1.063
    pid_Kd: 108.982
    min_temp: 0
    max_temp: 250

    [tmc2209 extruder]
    uart_pin: PC11
    microsteps: 16
    run_current: 0.650
    hold_current: 0.500
    stealthchop_threshold: 5

    [heater_bed]
    heater_pin: PC9
    sensor_type: ATC Semitec 104GT-2
    sensor_pin: PC3
    control: pid
    pid_Kp: 54.027
    pid_Ki: 0.770
    pid_Kd: 948.182
    min_temp: 0
    max_temp: 130

    [fan]
    pin: PA8

    [mcu]
    serial: /dev/serial/by-id/usb-Klipper_Klipper_firmware_12345-if00

    [printer]
    kinematics: cartesian
    max_velocity: 300
    max_accel: 3000
    max_z_velocity: 5
    max_z_accel: 100

    [static_digital_output usb_pullup_enable]
    pins: !PC13

    [board_pins]
    aliases:
        # EXP1 header
        EXP1_1=PB5, EXP1_3=PA9,   EXP1_5=PA10, EXP1_7=PB8, EXP1_9=<GND>,
        EXP1_2=PB6, EXP1_4=<RST>, EXP1_6=PB9,  EXP1_8=PB7, EXP1_10=<5V>
```

NOTE: This is the default config included in klipper (~/kilpper/config/generic-bigtreetech-skr-mini-e3-v1.2.cfg), if you use a different extruder, calibrate your e-steps. If you are coming from Marlin, it's pretty much `1/your marlin e-steps`. Divide 1 by your marlin e-steps then round it up to six decimal places then put it in to `step_distance` in the `[extruder]` section above.

If you want the LCD to be functional add this to the bottom of the printer.cfg (after `[board_pins]` section).

```
    [display]
```

```
lcd_type: st7920
cs_pin: EXP1_7
sclk_pin: EXP1_6
sid_pin: EXP1_8
encoder_pins: ^EXP1_5, ^EXP1_3
click_pin: ^!EXP1_2
#kill_pin: ^!EXP2_8

[output_pin beeper]
pin: EXP1_1
```

Again, this is for stock ender 3 LCD, if you use something else use the pin map above as a guide to your LCD. Common LCD configs are listed in ~/kilpper/config/sample-lcd.cfg but sample st7920 config does not work with skr mini e3, you have to use the one above.

At this point, run the auto hot end and auto heated bed pid tuning options in the menu, it takes a while but make sure you go through them.

=====

Also there are some test menus and delta settings and what not in the default menu, the config below removes these and adds LCD Corner Leveling to prepare menu:

```
[bed_screws]
screw1: 32,32
screw2: 203,32
screw3: 203,203
screw4: 32,203
screw5: 117,117

[menu __main]
type: list
name: Main
items:
    __tune
    __control
    __temp
    __filament
    __prepare

[menu __prepare]
type: list
enable: !toolhead.is_printing
name: Prepare
items:
    .__bed_leveling
    .__hotend_pid_tuning
    .__hotbed_pid_tuning
    .__host_restart
    .__firmware_restart

[menu __prepare __bed_leveling]
type: list
enable: !toolhead.is_printing
name: Level Corners
items:
    .__home
    .__start
    .__accept
    .__adjust
    .__abort_screws

[menu __prepare __bed_leveling __home]
type: command
name: Home All
gcode: G28
enable: !toolhead.is_printing

[menu __prepare __bed_leveling __start]
type: command
name: Start
enable: !toolhead.is_printing
gcode: BED_SCREWS_ADJUST

[menu __prepare __bed_leveling __accept]
type: command
name: Accept
enable: !toolhead.is_printing
```

```
    gcode: accept

[menu __prepare __bed_leveling __adjust]
type: command
name: Adjusted
enable: !toolhead.is_printing
gcode: adjusted

[menu __prepare __bed_leveling __abort_screws]
type: command
name: Abort
enable: !toolhead.is_printing
gcode: abort

[menu __filament __feed]
type: input
name: Feed: {0:.1f}
parameter: 0
input_step: 1.0
gcode:
    M83
    G1 E{0:.1f} F100
```

Bed Screw/Corner/Paper Leveling works like this:

- Go to Prepare -> Level Corners
- First <Home All>
- Then <Start>, extruder will move to first screw
- Adjust your screw with paper test
  - Click <Accept> if you only did a minor adjustment
  - Click <Adjusted> if you did a bigger adjustment that(more than quarter of turn of the bed screw)
    - Klipper will remember the "adjusted" points and will make you level corners that might have gone off because of your adjustment
- extruder will move to the next screw
- repeat until it's done, or stop the leveling process with <Abort>

You can use the above examples to create your own menus, you can pretty much run any g-code you want by clicking a menu item.