Mini-Doku: BTT TFT35 V3.0.1 + Raspberry Pi 5 mit Klipper

Ziel

Ein BTT TFT35-Display direkt per UART am Raspberry Pi 5 betreiben - ohne Octopus oder MCU-UART, vollstndig ber Linux-Serialkommunikation mit Klipper.

Hardware

- Raspberry Pi 5 (mit MainsailOS / Debian Bookworm)
- BTT TFT35 V3.0.1 Display
- UART-Verbindung: TX/RX/GND vom Pi (GPIO14/15 -> ttyAMA0)
- Baudrate: Display auf 115200 gesetzt

1. printer.cfg - Add-on-Konfiguration

```
[tftbridge]
tft_device: /dev/ttyAMA0
tft_baud: 115200
tft_timeout: 0
klipper_device: /home/nick/printer_data/comms/klippy.serial
klipper_baud: 250000
klipper_timeout: 0
```

2. Python-Skript (tftbridge.py)

```
import serial
import threading
import time

def load_config(config):
    tft_serial = serial.Serial(
        config.get("tft_device", "/dev/ttyAMA0"),
        config.getint("tft_baud", 115200),
        timeout=config.getfloat("tft_timeout", 0),
    )
    klipper_serial = serial.Serial(
        config.get("klipper_device", "/home/nick/printer_data/comms/klippy.serial"),
        config.getint("klipper_baud", 250000),
        timeout=config.getfloat("klipper_timeout", 0),
    )

    threading.Thread(target=tft2klipper, args=(tft_serial, klipper_serial), daemon=True).start()
    threading.Thread(target=klipper2tft, args=(tft_serial, klipper_serial), daemon=True).start()
    return {}

def tft2klipper(tft_serial, klipper_serial):
    while True:
        try:
            line = tft_serial.readline()
            if line:
                klipper_serial.write(line)
```

```
        except Exception:
            pass
        time.sleep(0.01)

def klipper2tft(tft_serial, klipper_serial):
    while True:
        try:
            line = klipper_serial.readline()
            if line:
                tft_serial.write(line)
        except Exception:
            pass
        time.sleep(0.01)
```

Warum time.sleep(0.01)?

Ohne diese Pause steigt die CPU-Auslastung auf uber 40 % im Leerlauf. Mit sleep(0.01) bleibt sie unter 1 % - selbst beim Homen oder Drucken.

Live-Ergebnisse

- CPU-Last im Standby: 0.09 %
- CPU-Last beim Homen (Z-Achse): max. 0.26 %
- TFT reagiert zuverlassig
- Kein MCU-Shutdown ("Timer too close")
- Filamentsensor funktioniert einwandfrei