HOW TO DEVELOP AND TEST A PWM DRIVER

ABOUT ME

- Senior Software R&D Engineer @ Baylibre
- PWM subsystem maintainer
- contributor to various kernel subsystems
- ukleinek on libera and OFTC
- PGP: E2DCDD9132669BD6

... AND BAYLIBRE

- Embedded software consultancy based in Nice, France
- ~60 engineers
- various OSS projects (Linux, U-Boot, Zephyr, ...)

MOTIVATION FOR THIS TALK

- Review of new drivers takes long
- What to you want for the PWM subsystem?

WHAT IS A PWM?

- periodic square wave signal
- used to
  - blink or dim LEDs
  - drive display backlights
  - motor control (e.g. fan)
  - remote controls

WHAT IS A PWM?

- period + duty cycle + duty_offset [ns]

LOWLEVEL CALLBACKS

```
int (*round_waveform_tohw)(struct pwm_chip *chip,
                           struct pwm_device *pwm,
                           const struct pwm_waveform *wf,
                           void *wfhw);
int (*round_waveform_fromhw)(struct pwm_chip *chip,
                             struct pwm_device *pwm,
                             const void *wfhw,
                             struct pwm_waveform *wf);
int (*read_waveform)(struct pwm_chip *chip,
                     struct pwm_device *pwm,
                     void *wfhw);
int (*write_waveform)(struct pwm_chip *chip,
                      struct pwm_device *pwm,
                      const void *wfhw);
```

ROUNDING RULES WAVEFORM -> HARDWARE

- pick period, duty cycle and duty offset rounding down
- round up if needed (and return 1)

Enable `CONFIG_PWM_DEBUG`

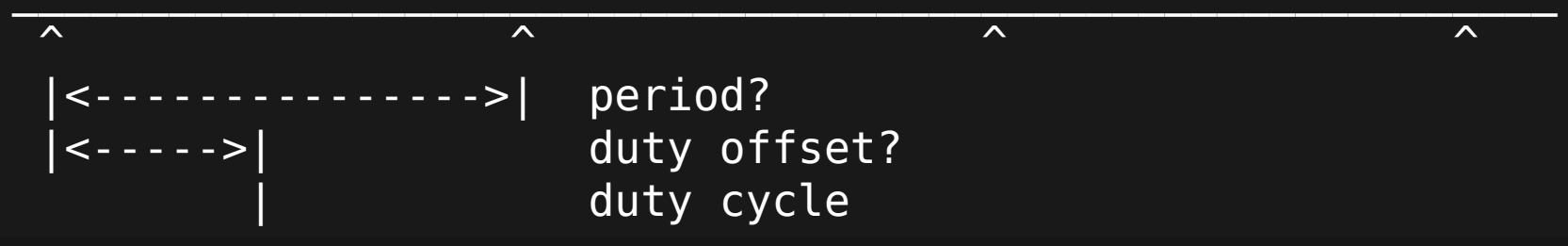
ROUNDING RULES HARDWARE -> WAVEFORM

- round up all parameters

AMBIGUITIES FOR SELECTING PARAMETERS

AMBIGUITIES FOR SELECTING PARAMETERS

- Not all different representations are distinguishable in reality



AMBIGUITIES FOR SELECTING PARAMETERS

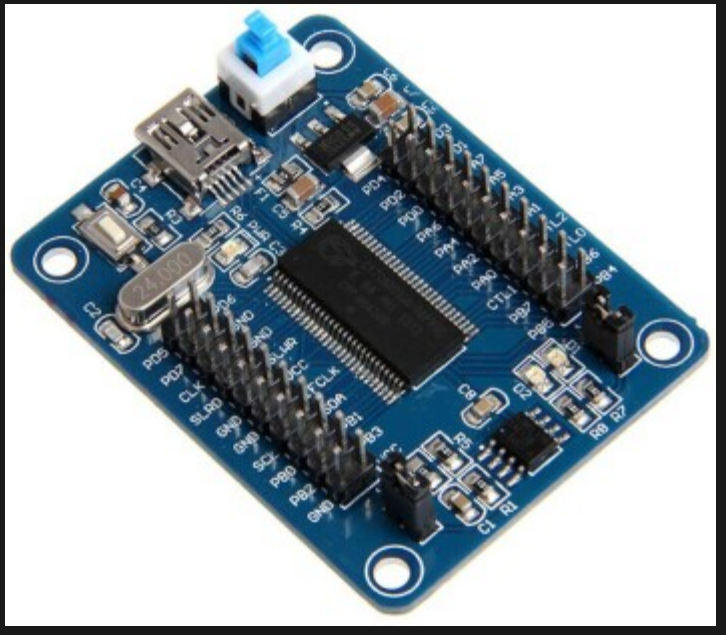
- Respect artificial period
- consistent behaviour

HARDWARE FOR TESTING

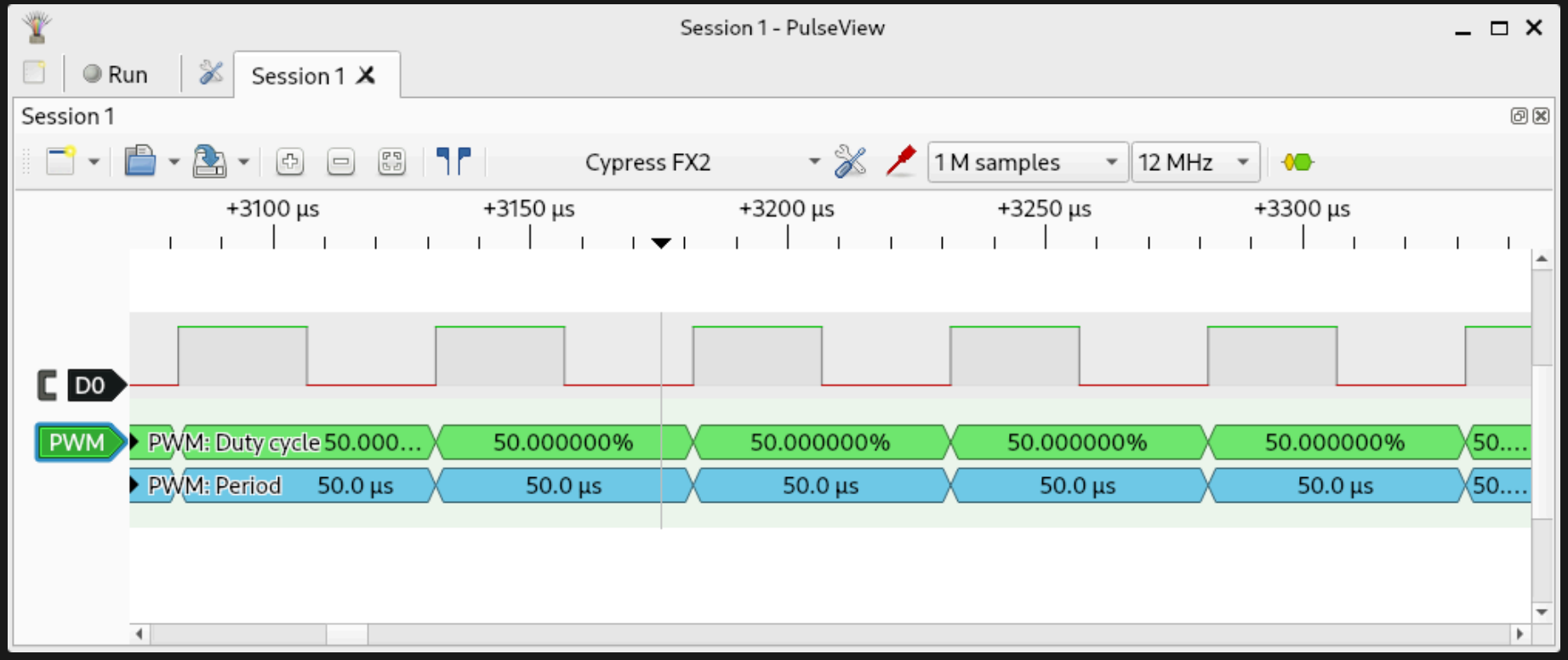
sigrok/pulseview =>
https://sigrok.org/wiki/Supported_hardware

I personally have Lcsoft Mini board, ca. 8 € on ebay
(search term: CY7C68013A)

HARDWARE FOR TESTING



SIGROK/PULSEVIEW



TOOLS

- libpwm
- `/sys/kernel/debug/pwm`
- memtool (devmem2)

STRATEGIES

- Test corner cases (min/max period, 0% / 100% relative duty cycle)
- Test apposed settings (`pwmtestperf`)
- Use memtool to see how PWM behaves in the middle to changing configuration
- Use slow clock rate in the provider, high sample rate on capture side

FUTURE?

- Couple PWM channels
- Instrument GPIO for PWM state transitions
- only n periods

ADS

EMBEDDED RECIPES De Nice

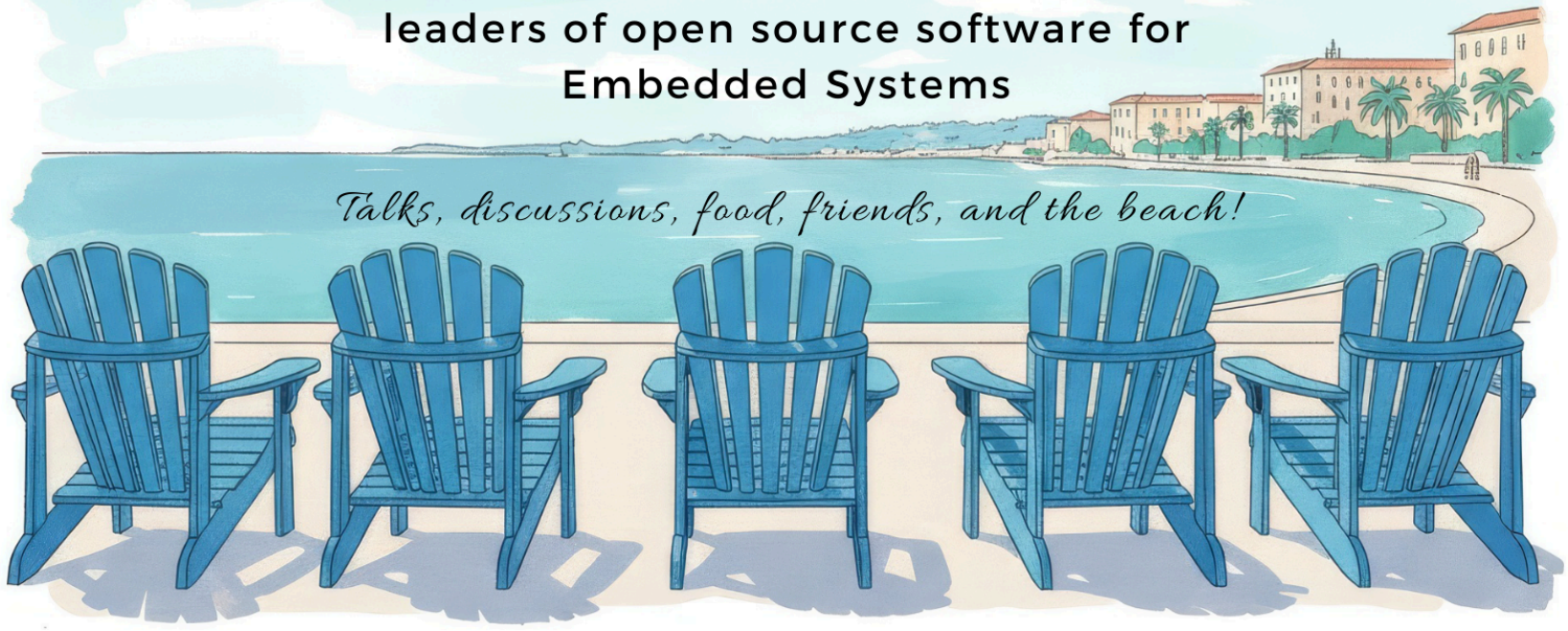
by Baylibre

2026 May 27-28

Nice, France

Join the core developers and thought leaders of open source software for Embedded Systems

Talks, discussions, food, friends, and the beach!



CFP

Home

Matrix Space

ADS

We 🤍 open-source, so does our partner >commit

>commit is a VC fund investing in early-stage Commercial Open-Source Startups with European roots.

Companies backed by >commit & its team

Twenty · UMA · Mastra · White Circle · Keep · Better-Auth

Graphcore · Pangolin · Pyannote · Pandas AI · Stealth project

Disclaimer: BayLibre SAS is a Limited Partner in the >commit fund

Learn more at commit.fund

THANKS & Q & A