Rapid prototype development using Zephyr

A product developer’s perspective

Artur Nidens

Freelancer Embedded Software

The prototypes

Multi sensor transmitter for process industry

- New design shall be accessible via smartphone
- Host MCU handles core services but has no BLE capability

→ Peripheral MCU needed for BLE

IOT gateway for water utility

- Analog sensor in remote location without grid and wired network

→ Digitizing analog sensor data
→ Battery powered
→ Cellular gateway

The prototypes

Multi sensor transmitter for process industry

- New design shall be accessible via smartphone
- Host MCU handles core services but has no BLE capability

→ Peripheral MCU needed for BLE

IOT gateway for water utility

- Analog sensor in remote location without grid and wired network

→ Digitizing analog sensor data
→ Battery powered
→ Cellular gateway

I didn’t chose Zephyr, Zephyr chose me

The workflow

Pick hardware:
nRF52833
nRF9160

Find closest sample:
NUS
MQTT

Modify sample

Some useful modules

ZBus

- Lightweight and flexible software bus enabling a simple way for threads to talk to one another in a many-to-many way

ZTest

- Test framework for tests simulated or ran on hardware
- Simulation is handy but does not always reflect reality (Soft device controller vs. Zephyr BLE controller)

Pros and cons

Linux inspired parts like device tree, Kconfig	Very generic, used in many other projects	Feels foreign if not used to linux
Resources (docs, samples, tutorials, etc)	Plenty information on most needs	Resources not in one place, different responsibilities

Final thoughts

Feels complex upfront but

- **Leverage samples, subsystems, drivers**
- **Don’t fear not knowing entire stack**
- **→ Both prototypes ready for pilot installation within approx. 3 months**

Thank You