SAMSUNG Research

## Netdev 0x14

# IoT Networking Workshop

- June 16, 2020

- Samsung Research UK
Stefan Schmidt <s.schmidt@samsung.com>

# Agenda

- 15:00 RPL: IPv6 Routing Protocol for LLNs
- 15:30 Workshop welcome
- 15:40 ieee802154 & 6lowpan update
- 16:00 CAN subsystem update
- 16:20 Discussion and sum up

# Welcome

# Welcome

## Introduction

- Linux-wpan
- Ieee802154 & 6lowpan
- rpld
- Alex & Stefan

- Linux CAN
- Marc & Oleksij

- Who else is here?
- What are you working on?
- What are you expecting from this workshop?

# ieee802154 & 6lowpan update

# Website

- New website launched

**wpan.cakelab.org**

- Old page was hard to maintain
- Only Alex had access

**https://linux-wpan.org**

- New page built with jekyll and github pages
- Easy git access
- Free hosting
- Well known pattern

- Existing content moved over and redirected
- More content need to be created (rpld, Pi setup)
- Better integration with in-tree Kernel docs

# Wpanusb

- Generic USB device driver

## Past

- Developed by Andrei Emeltchenko @intel
- ieee802154 does not specify an HCI
- Own USB interface spec based on atusb
- Developed together with Zephyr firmware
- Project stalled in 2018

## Present

- Andrei has no time
- Taken over by Koen, Eric and Stefan now
- Extend USB interface spec to be more flexible
- e.g. capabilities provided by firmware, different frequency bands, power levels, permanent address, etc
- RIOT OS firmware developed in parallel

- Zephyr firmware needs to be updated to new spec
- RIOT firmware needs testing with more boards and radios
- Bare metal or Contiki implementations?

# Linux CAN update

# SocketCAN and J1939

Oleksij Rempel – ore@pengutronix.de
Marc Kleine-Budde – mkl@pengutronix.de

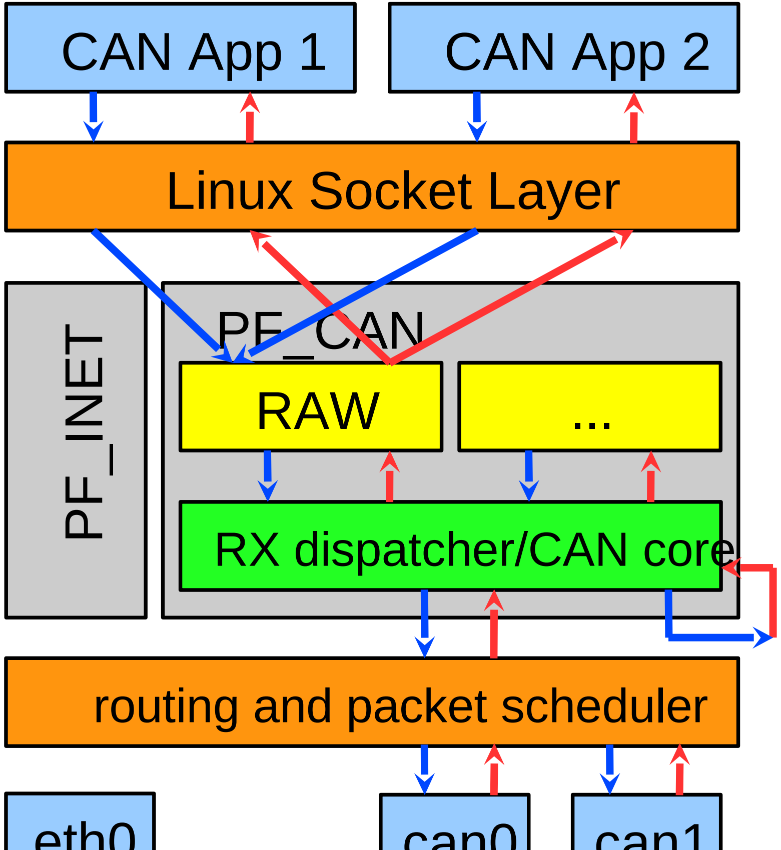
# … spend some words on CAN

- CAN != Ethernet
- 2 wire cable
- broadcast medium
  - CSMCA (Carrier Sense Multiple Collision Avoidance)
- multi master bus
  - 100base-T1 is single master bus, 10base-T1s multi drop bus
- Speed:
  - up to up to 1 Mbit/s (CAN-2.0)
  - 8...12 Mbit/s (CAN-FD, CAN-XL)

# … spend some words on CAN

- 11 or 29 bit address (CAN-ID)
- prioritization of CAN frames by CAN-ID
- Payload:
  - 8 Bytes (classic CAN-2.0)
  - 64 Bytes (CAN-FD), CAN-FD is compatible with existing CAN 2.0 networks.
  - 2048 Bytes (CAN-XL, SiG is "near completion", next step ISO standardization)
    - https://www.bosch-semiconductors.com/news/t-newsdetailpage-4.html
    - https://can-newsletter.org/engineering/engineering-miscellaneous/200103_2020s-decade-welcome-can%20xl_cia

# SocketCAN isn‘t Rocket Science!



- HW abstraction for different controllers
- Currently supported socket types: CAN_RAW, CAN_BCM, CAN_J1939
- CAN-2.0 and CAN-FD - seamless integration, configurable on socket level

# What is SAE J1939?

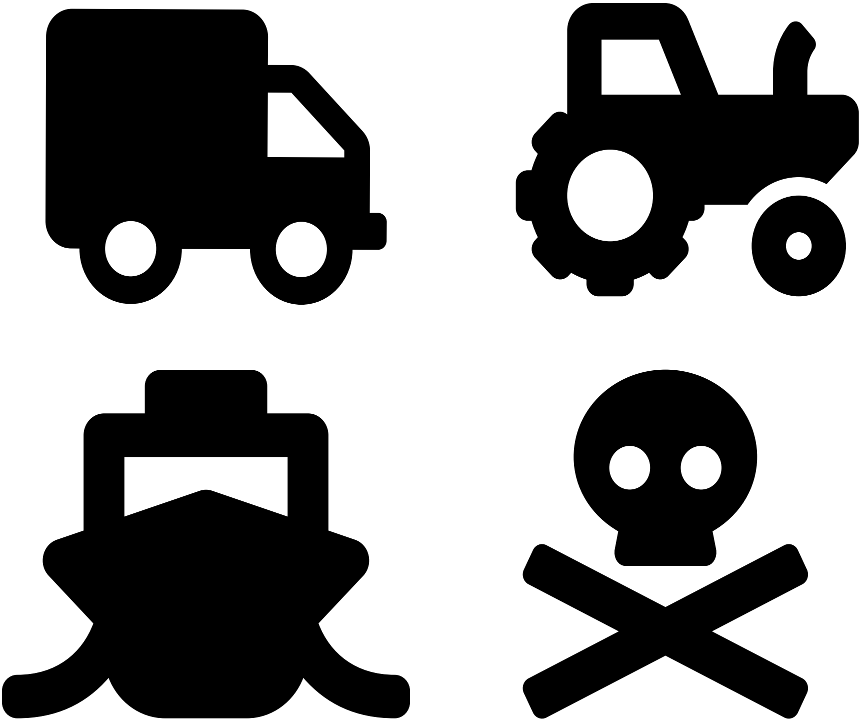
- Recommended practices and standards:
  - SAE J1939: for truck and bus applications, possible other..
    - ISO 11783: for tractors and machinery for agriculture and forestry
    - NMAE 2000: marine sensors and display units within ships and boats
    - MilCAN: military vehicles
- Reliable transfer for large amounts of data
- SAE J1939: Transport Protocol = 1785 Bytes (255 packets * 7 Bytes/packet)
- ISO 11783: Extended Transport Protocol ~ 112 MiB (16.777.215 packets * 7 Bytes/packet)

# CAN_J1939 socket

- Available since kernel 5.4 (Nov 2019)
- 20 bug reports provided by google syzkaller, 17 are already fixed.
- Last months we see increased activity with questions and reports on mailing lists and issues trackers.
- Please tell us your experience!

# SAE J1939 Linux Kernel Implementation

- Should be able to cover:
  - SAE J1939
  - ISO 11783
  - NMEA2000
  - MilCAN A



# CAN_BCM socket

- Broadcast manager
- Available since kernel 2.6.24 (Nov 2007)
- TX: periodic sending of unchanged CAN frames "I'm still alive"
- RX: monitoring of periodic reception, notification on timeout or value change

# TODO

- J1939
  - Proper way to export address claiming cache to the user space
  - Quirky buses.
  - Test automation (follow osmocom testing experience?)
- Time Sensitive Networking (TSN)
  - proper time stamps
  - TT-CAN (Time Triggered-CAN), for example in m-can (IP core on stm32-mp1)
- PHYs
  - investigate if it is possible to use (Ethernet-)PHY framework
  - following functionalities are currently expected on CAN PHYs:
  - switch PHY on/off
  - listen only support
  - wake-up support
  - cable testing
- HW filter support

Thank you!

Questions?

# Discussion and sum up

# Thank You