Achieving 10 year battery life with RIOT

Benjamin Valentin

ML!PA Consulting GmbH

September 15, 2020

Section 1

Motivation

Use Case

Situation

- IoT Toolbox: Customer specific sensor solutions
- Sensors installed in hard to reach places
- Rugged enviroments
- No Maintenance

Solution

- battery powered sensors
- radio communication (sub-GHz 802.15.4g)

Disclaimer

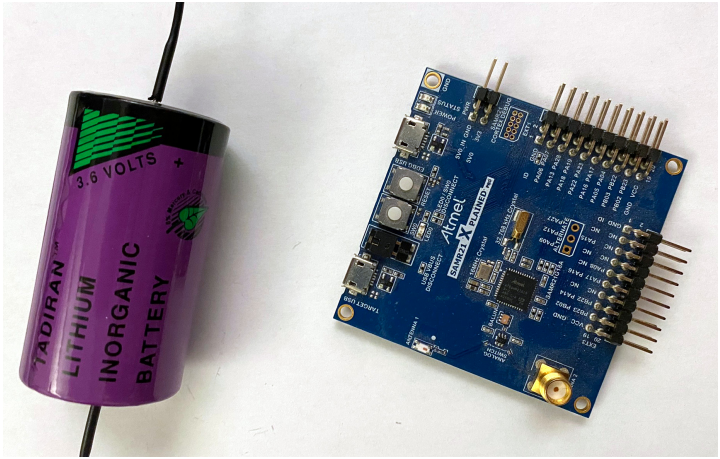
- There are differences between platforms
- I'm not an electrical engineer
- Don't apply changes without measuring the effect!

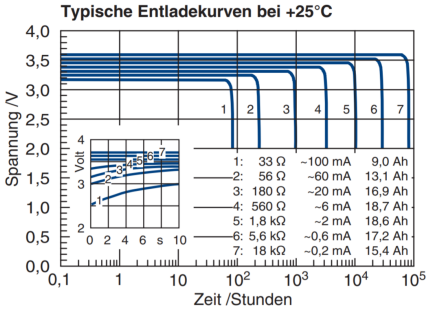
Section 2

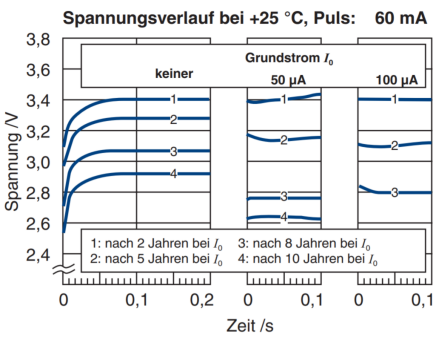
Start with a large battery

Start with a large battery

- D-cell sized
- low self-discharge
- ~ 15Ah →up to 100 µA continuous current for 10 years
- with 50 µA still 2.6V after 10 years







Section 3

Know your components

Get to know your MCU

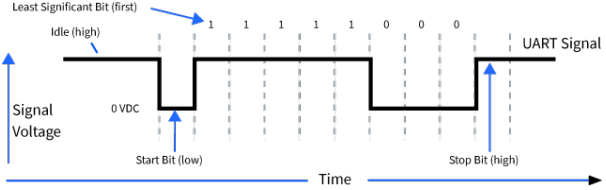
- read the Data Sheet *and* the Errata Sheet
  - Buck Converter not usable with fast internal clocks (DPLL, DFLL) on `sam0`
  - voltage regulator selection presistent across reflashes! (not cold boots)
  - use an external Oscillator
- No EXTI in Deep Sleep, only 5 RTC / Tamper pins
- Reset on wake-up

SAM D51

- Cortex-M4 with FPU
- actually not a *low power* part
- 1.71 V to 3.63 V
- 3.3 µA in BACKUP mode (8k backup RAM retained, RTC)
- 149 µA/MHz in ACTIVE mode with LDO
- 73 µA/MHz in ACTIVE mode with buck converter

Get to know your MCU

- GPIO pins keep their configuration / level in Deep Sleep
  - Reconfigure pins if the connected peripheral does not to avoid voltage differential
  - don't leave pins with external pull-up LOW or pins with external pull-down HIGH
    $3.3V / 10\ k\Omega = 330\ \mu A$
- Keep in mind IDLE-HIGH signals like UART



- Don't measure with Debug UART attached
  - UART adapter draws power via TX line
  - OR board powered via Debug UART

Take a look at other components on the PCB

- Make sure to put peripherals to sleep or cut the power
- don't wake sleeping peripherals if they are not used
- $\sim$ 25 µA during Deep Sleep (of 50 µA battery budget)
- $\rightarrow 25\mu A * 10 * 365 * 24 * 60 * 60s = 7884As$ left
- $\rightarrow 7884As/(10 * 365 * 60s) = 36mA$ for a minute each day

AT86RF215

- 1.8 V to 3.6 V
- 3 mA in TRXOFF (reset) state
- 30 nA in SLEEP state
- 18.6 mA in RX mode with MR-O-QPSK & Reduced Power Consumption

LIS2DH12

- accelerometer / shock sensor
- 6 µA low-power mode

MTK3333

- GNSS
- 9 µA backup node

Section 4

Write a low power Application

Stay awake as little as possible

- CPU will reset with each period, keep persistent data in `BACKUP_RAM`
- Take a measurement
- Only initialize the radio when needed (`gnrc_netif_init_devs` instead of `auto_init_gnrc_netif`)
- Wait for `GNRC_IPV6_EVENT_ADDR_VALID` event (got prefix from border router)
- send the data
- shut down the radio
- shut down the sensors
- sleep till next measuring period

Section 5

Questions?