POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Course: Internet of Things - 5 CFU

Professor: Matteo Cesana
Students: Bruno Arena, Matteo Cavalleri
Personal IDs: 10806605, 10768765

Homework – Exercise 2

POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Abstract

Redesign of an IoT system for indoor localization of dental prostheses in a nursing home, based on BLE beacons embedded in each denture and gateways estimating position via RSSI measurements. The current firmware transmits an advertising packet every 10 s unconditionally, resulting in constant energy consumption even when localization is not actually required.

Answer

The original firmware transmits a periodic BLE advertising packet every 10 s regardless of the denture state. This wastes energy because, for most of the day, the denture is either in use, stored in a known place, or not actively searched by caregivers. The key idea is to make the beaconing event-driven and context-aware: the denture remains in deep sleep most of the time, sends very sparse heartbeat advertisements to maintain basic presence information, and temporarily increases the advertising rate only when motion or a localization-relevant event is detected. Motion is not treated as a direct indication that the denture is lost, but as a low-power trigger to wake up the system and decide the proper operating mode.

Proposed architecture.

1. *Beaconing logic*: three operating regimes are used.

    a. Deep idle / heartbeat mode: when no motion is detected and the denture is assumed to be in a known or non-critical state, the device remains in deep sleep and transmits only a sparse heartbeat advertisement, for example one beacon every 30 minutes. This provides basic presence information while minimizing radio usage.
    b. Motion event mode: when the accelerometer detects movement, the BLE SoC wakes up and transmits advertisements at a moderate rate, for example one beacon every 10 s for a limited observation window of 2 minutes. This allows the gateways to update the denture position when it has likely been moved, without continuously using a high advertising rate.
    c. Search mode**:** the tag periodically performs a short BLE scan window (10 ms every 30 s) listening for a search-trigger packet emitted by the nearest gateway. When the backend marks the denture as 'to be located', the gateways start broadcasting the trigger; at the next scan window the tag receives it and increases the advertising rate to one beacon per second for up to 5 minutes.

B. *Additional sensors*: 3-axis MEMS accelerometer with hardware wake-on-motion function (consumes < 1 μA while monitoring). Optionally, a temperature/contact sensor to distinguish "in use" from "lost" (the denture in the mouth is warm; sitting unused, cold → light beaconing even without motion).

<u>*Pseudocode*</u>

```
initialize_BLE()

initialize_accelerometer_wake_on_motion(threshold = Tmov)

initialize_optional_temperature_or_contact_sensor()

configure_heartbeat_timer(period = 30 minutes)

 state = DEEP_IDLE

 loop forever:

    enter_deep_sleep()

    if wakeup_source == HEARTBEAT_TIMER:

      send_BLE_beacon()

      state = DEEP_IDLE

    if wakeup_source == MOTION_INTERRUPT:

      read_accelerometer()

      read_optional_temperature_or_contact_sensor()

       if denture_probably_in_use():

         send_BLE_beacon()

         state = DEEP_IDLE

       else:

         state = MOTION_EVENT
```

```
        start_timer(duration = 2 minutes)
        while timer_not_expired():
            send_BLE_beacon()
            sleep(10 seconds)
        state = DEEP_IDLE
on scan_window_timer (every 30 s):
    rx = ble_scan(window = 10 ms)
    if rx contains SEARCH_TRIGGER:
        state = SEARCH_MODE
        start_timer(duration = 5 minutes)
        while timer_not_expired():
            send_BLE_beacon()
            sleep(1 second)
state = DEEP_IDLE
```

- *Commercial components*: SoC Nordic nRF52810 / nRF52832 (BLE 5, sleep ~1 µA with RTC), accelerometer ST LIS2DW12 or Bosch BMA400 (wake-on-motion < 1 µA, programmable threshold). All in small packages suitable for a denture.

- *Energy estimate*. We assume that localization is required on average for 1 hour per day. During this time, the tag transmits one BLE advertisement every 10 s, which is the same advertising interval used by the original firmware. During the remaining 23 hours, the denture is assumed to be either in use or stored in a known place, so the tag only sends sparse heartbeat advertisements every 30 minutes.

The original firmware sends one BLE advertisement every 10 s for the whole day:

$$N_{old} = 24 * \frac{3600}{10} = 8.640 \frac{beacons}{day}$$

Proposed event-driven solution:
During the 1 hour/day in which localization is required:

$$N_{loc} = 1 * \frac{3600}{10} = 360 \frac{beacons}{day}$$

During the remaining 23 hours, with one heartbeat beacon every 30 minutes:

$$N_{idle} = 23 * \frac{3600}{1800} = 46 \frac{beacons}{day}$$

$$N_{search} = 2 * 300 = 600 \frac{beacons}{day}$$

Therefore, the total number of daily advertisements is:

$$N_{new} = N_{loc} + N_{idle} + N_{search} = 360 + 46 + 600 = 1.006 \frac{beacons}{day}$$

The estimated reduction in the number of BLE advertisements is:

$$Saving = 1 - \frac{N_{new}}{N_{old}} = 1 - \frac{1.006}{8.640} = 88\%$$

Assuming a CR2032 coin cell with nominal capacity 220 mAh @ 3 V and typical self-discharge of 1 %/year, the lifetime is bounded by the dominant term among:

- radio transmissions: $N_{new} * E_{tx}\ per\ day$
- deep-sleep current: $I_{sleep} * V * 86400 \frac{s}{day} = 0{,}26 \frac{mWh}{day}$
- self-discharge: $\approx 6.6 \frac{mWh}{year}$

Even with the original firmware, the radio consumption is smaller than the SoC sleep current.
The 1-year lifetime quoted in the original specification is therefore likely limited by hardware choices (SoC with high quiescent current, no power gating) rather than by radio activity.

The proposed firmware reduces radio energy ten times but the real benefit comes from architectural improvements (deep sleep + wake-on-motion + scarce heartbeat) that allow the SoC to spend > 99.9 % of the time in true low-power mode, extending the lifetime to 5–8 years.