Research Article

System-Level Power Consumption Analysis of the Wearable Asthmatic Wheeze Quantification

Dinko Oletic (ID) **and Vedran Bilas** (ID)

Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia

Correspondence should be addressed to Vedran Bilas; vedran.bilas@fer.hr

Long-term quantification of asthmatic wheezing envisions an m-Health sensor system consisting of a smartphone and a body-worn wireless acoustic sensor. As both devices are power constrained, the main criterion guiding the system design comes down to minimization of power consumption, while retaining sufficient respiratory sound classification accuracy (i.e., wheeze detection). Crucial for assessment of the system-level power consumption is the understanding of trade-off between power cost of computationally intensive local processing and communication. Therefore, we analyze power requirements of signal acquisition, processing, and communication in three typical operating scenarios: (1) streaming of uncompressed respiratory signal to a smartphone for classification, (2) signal streaming utilizing compressive sensing (CS) for reduction of data rate, and (3) respiratory sound classification onboard the wearable sensor. Study shows that the third scenario featuring the lowest communication cost enables the lowest total sensor system power consumption ranging from 328 to 428 μW. In such scenario, 32-bit ARM Cortex M3/M4 cores typically embedded within Bluetooth 4 SoC modules feature the optimal trade-off between onboard classification performance and consumption. On the other hand, study confirms that CS enables the most power-efficient design of the wearable sensor (216 to 357 μW) in the compressed signal streaming, the second scenario. In such case, a single low-power ARM Cortex-A53 core is sufficient for simultaneous real-time CS reconstruction and classification on the smartphone, while keeping the total system power within budget for uncompressed streaming.

1. Introduction

Asthma is a chronic respiratory disease that affects more than 300 million patients [1]. One of its specific symptoms is the occurrence of so-called "asthmatic wheezing" in respiratory sounds, caused by constriction of bronchial airways [2, 3]. Asthmatic wheezing is assessed by auscultation of either patient's chest, back, or neck [4, 5]. Being noninvasive and not requiring patient cooperation, quantification of wheezing proves beneficial as an independent method for diagnosis and continuous monitoring of asthma [6]. It is especially suitable for diagnosis of asthma in children and nocturnal asthma [7, 8], where traditional spirometry-based methods of assessment of respiratory function parameters are not appropriate (i.e., home peakflowmetry). Automated quantification of wheezing is performed on respiratory sounds recorded by a single acoustic sensor [4, 9]. It entails

identification of an unknown number of intermittently appearing, temporally evolving frequency lines, embedded in respiratory noise [4]. Algorithm implementing this processing-intensive task most commonly combines spectrotemporal features drawn from the short-term Fourier transform (STFT) [10–14], Mel-frequency cepstral domain (MFC) [15, 16], wavelet transform [17, 18], empirical mode decomposition [19], and a variety of classification schemes, including decision trees [10, 12], neural networks [18], and support vector machine [13–15, 18]. Detailed reviews given in [12, 18, 20] report classification performance ranging on average from 90 to 95% of sensitivity and specificity. Building upon the existing analysis of low-power DSP implementable wheeze detection algorithms [12, 21], we fix the algorithm choice to the STFT frequency line tracking algorithms based on either the empirical rules [12] or the hidden Markov model (HMM) [22, 23]. Both yield similar,
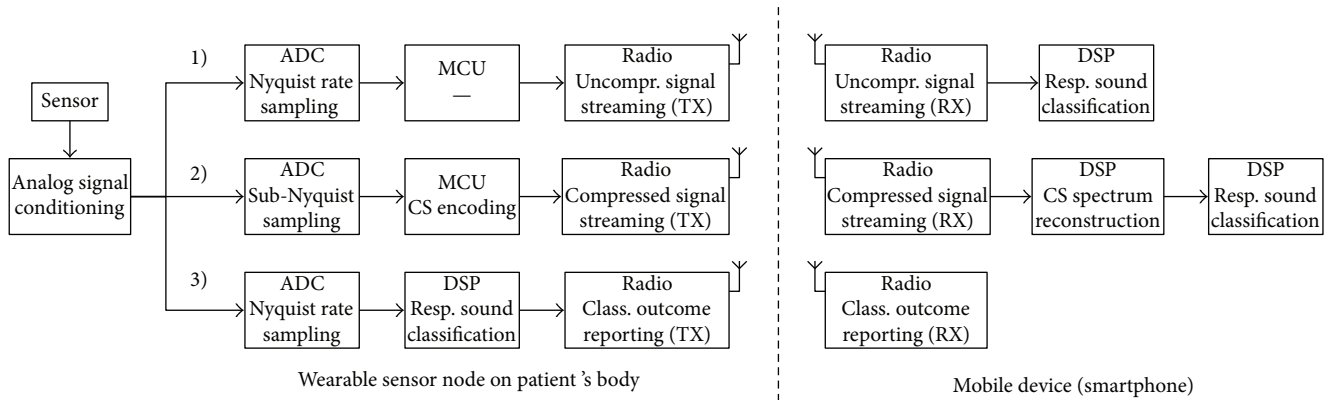
FIGURE 1: Analyzed operating scenarios: (1) raw, uncompressed signal streaming (referent), (2) streaming of CS-encoded signal, and (3) classification onboard the wearable sensor.

representative classification performance of 87–89% sensitivity and 93–96% specificity.

Commonly envisioned concept of an m-Health sensor system for quantification of asthmatic wheezing consists of a smartphone and a wireless acoustic sensor [24–26]. In recent years, some commercial products have appeared on the market, featuring the smartphone-based electronic asthma diary application, accompanied by the wheeze quantification sensor in the form-factor of a hand-held on-demand measurement device [27]. In order to allow continuous patient monitoring, current research efforts are aimed towards enabling construction of wearable (body-worn) wheeze quantification sensors [12, 13, 16, 17, 28], consisting of the following subsystems: acoustic transducer (sensor), analog signal conditioning circuit, A/D converter, a power-efficient digital signal processing unit, and a low-power digital radio module for communication with the smartphone.

Being battery powered, both the wearable sensor and the smartphone are power-constrained devices. Thus, the main criteria guiding the design of an m-Health asthma quantification sensor system are (1) minimization of power consumption and (2) retention of sufficient respiratory sound classification accuracy. Trade-off between mentioned criteria is affected by (1) the choice of respiratory sound classification algorithm, (2) the system architecture in terms of organization of signal acquisition, processing, and communication between the wearable sensor and smartphone, and (3) the characteristics of hardware components implementing each particular subsystem [29].

System-level analysis presented in this paper contributes by proposing how to power-efficiently organize a processing-intensive wheeze detection sensor system, consisting of a wireless wearable acoustic sensor and a smartphone. Specifically, we aim to quantify the universal trade-off between the power cost of acquisition, processing, and communication [29], for this specific processing-intensive application. Also, we provide generalized guidelines on hardware component architectures best fitting the application. The analysis builds up upon our extensive prior research on novel energy-efficient signal acquisition and wireless transport schemes [30], design of specialized low-power

wheeze recognition algorithms suitable for running onboard energy-constrained devices (sensor node and smartphone) [12, 23], and verification of all subsystems on several hardware laboratory prototypes [12, 22, 31, 32].

The paper compares total power requirements of the sensor system in three different operating scenarios shown in Figure 1. In the first, referent operating scenario, the sensor acquires the signal at Nyquist rate. Apart from signal acquisition, no particular DSP processing tasks are performed on the sensor. Raw signal is streamed uncompressed via low-power radio communication interface over to the smartphone, where the respiratory sound classification algorithm is executed [24, 26]. The scenario is motivated by the idea of simplification of the sensor design and using the smartphone as the main signal processing platform, in order to simplify development and maintenance of respiratory sound classification software.

In order to lower the data rate and consequently the average power required for streaming, the signal from the sensor to a smartphone, the second scenario, exploits the compressibility of respiratory sounds in the frequency domain [30, 33] and applies a concept of compressed sensing (CS) [34, 35]. By combining signal acquisition and compression in a single linear transformation step (i.e., CS encoding), CS in comparison to classic audio coding techniques [36, 37] enables simultaneous lowering of the data rate, while retaining the low complexity of signal processing on the sensor. Here, the CS encoding on the sensor is performed by pseudorandomly subsampling the signal [31], effectively compressing it. The compressed signal is streamed over to the smartphone. By knowing the subsampling pattern, the CS decoder subsystem block obtains a reconstruction of the compressible STFT spectra from the CS-encoded signal [30, 32]. Finally, a robust hidden Markov model- (HMM-) based frequency line tracking algorithm [23] implemented on a smartphone [22] enables for wheeze detection from reconstructed spectra at less than 5% loss of classification accuracy.

In the third scenario, Nyquist rate signal acquisition and respiratory sound classification are performed onboard the wearable sensor, such as in [13, 16]. Here, we compare the processing burden of the DSP implementations of robust frequency line tracking based on HMM [22, 23] to the

TABLE 1: Comparison of acoustic sensors' power consumptions.

Technology	Component	Sensitivity	Imped	Power
MEMS microphone, digital (I²S)	ADMP441 [42]	−26 dBFS	—	2520 μW at 1.8 V
Electret condenser microphone	KEEG1542 [43]	−42 dB	2.2 kΩ	1000 μW at 2.0 V
Analog accelerometer	ADXL337 [44]	300 mV/g	32 kΩ	900 μW at 3.0 V
MEMS microphone, analog	ADMP404 [45]	−38 dBV	200 Ω	375 μW at 1.5 V
Accelerometer, digital (SPI)	ADXL345 [46]	3.9 mg/LSB	—	350 μW at 2.5 V
MEMS microphone, analog	ICS-40310 [47]	−37 dBV	4.5 kΩ	16 μW at 1.0 V

frequency-tracking algorithm mimicking the nearest neighbor association previously presented in [12], analogous to [10]. The sensor periodically reports the classification result to the smartphone. The scenario is aimed at minimizing the data traffic between the wearable sensor and the smartphone and making the wearable sensor independent with respect to radio link quality and smartphone-processing resources [38].

The paper is organized as follows: In Section 2, each of the sensor's subsystems is analyzed from the aspect of power efficiency. Based on this, total sensor power consumption is analyzed for each operating scenario in Section 3. In parallel, estimates for the power spent onboard the smartphone are given in Section 4. Finally, sensor system power consumption estimates are derived in Section 5. The paper is concluded in Section 6.

2. Power Analysis of the Wearable Sensor's Subsystems

2.1. Sensors and Analog Signal Conditioning. Requirements for design of an acoustic sensor and associated analog signal conditioning circuitry were derived from standardized guidelines for the respiratory sound acquisition [39–41]. First, microphones and accelerometers were evaluated as acoustic sensors. Microphones were chosen to exhibit flat frequency response in bandwidth of respiratory sounds (i.e., 100 to 1000 Hz) and accelerometers featuring a resonant frequency well above the upper corner frequency.

Representative acoustic sensor technologies complying with those requirements were evaluated: electret condenser microphone (KEEG1542, Knowles), MEMS microphone with analog output (ADMP404, Analog Devices; ICS-40310, Invensense), and MEMS microphone with digital output (ADM441, Analog Devices). Sensitivities of tested analog microphones ranging from −42 to −37 dB and of digital MEMS microphones were typically −26 dBFS (decibels relative to digitized full-scale reading). SNR varied from 50 to 62 dB. Capacitive MEMS accelerometers were chosen for evaluation (ADXL337, ADXL345, and analog devices).

Comparison in Table 1 shows that among tested sensor technologies, analog MEMS microphones feature the highest power efficiency. It is shown that power consumption as low as 16 μW may be obtained using ICS-40310. However, the common problem of low-powered analog MEMS microphones is high output impedance. Electret condenser microphones traditionally used in respiratory signal analysis exhibit significantly worse power efficiency than their MEMS counterparts. The tested model of the accelerometer, chosen

to fit into the power consumption budget of microphones, exhibited the drawback of lower bandwidth and lower sensitivity on its vertical axis, due to mechanics of the micromachined proof mass.

Advantage of the digital system-on-chip (SoC) acoustic sensors, such as ADMP441 or ADXL355, is integration of a complete signal chain, consisting of a microphone, audio amplifier, ADC, and standard encoding of digitized output signal (i.e., PDM, I²S, or SPI). However, this is paid with the high overall power consumption and the lack of flexibility (i.e., absence of a programmable amplifier and long wake-up time from power down).

Design of the analog signal conditioning circuit for the respiratory sound acquisition accommodates several functionalities. First is the signal amplification, as typical sensor's output signal magnitudes reside in range of 1 to 10 mV (as shown by typical sensitivities in Table 1). Special attention needs to be taken in case of MEMS microphones, as the amplifier's input is required to handle their output impedance typically in order of kΩ. Bandpass filtering with lower corner frequency of around 100 Hz is required, to filter out heart sounds. The upper corner frequency of the filter is adjusted according to the signal sampling frequency in order to prevent antialiasing. Assuming that the microphone model is chosen such to filter out the low frequencies by its frequency characteristics and that antialiasing is implemented by a passive RC filter, a single instrumentation amplifier may suffice for the signal conditioning requirements. Power consumption of such conditioning circuit, implemented by INA333 [48], is estimated to 85–95 μW by Spice simulation.

2.2. Signal Digitization. Two signal sampling scenarios were analyzed. The first is Nyquist rate signal sampling at sampling rates of 2 to 8 kHz, and the second is the compressive sampling (CS) at temporally nonuniform time instants, as proposed in [31]. In case of the CS, analog-to-digital converters (ADCs) were tested at sub-Nyquist sampling rates of 250 Hz to 1 kHz, corresponding to signal compression ratios of 8x to 2x. Based on the results from [31, 39] and [12], ADC resolution of 12 to 16 bits is required for the CS DFT/DCT spectrum reconstruction and the respiratory sound classification. Thus, the power efficiency tradeoffs of 12-, 16-, and 2-bit successive approximation (SAR) and sigma-delta ADCs were compared.

ADCs were assumed operating on demand (triggered by a signal processor) and entering power-saving state after completing each conversion (duty cycling). Thus,

TABLE 2: Parameters of the tested ADCs.

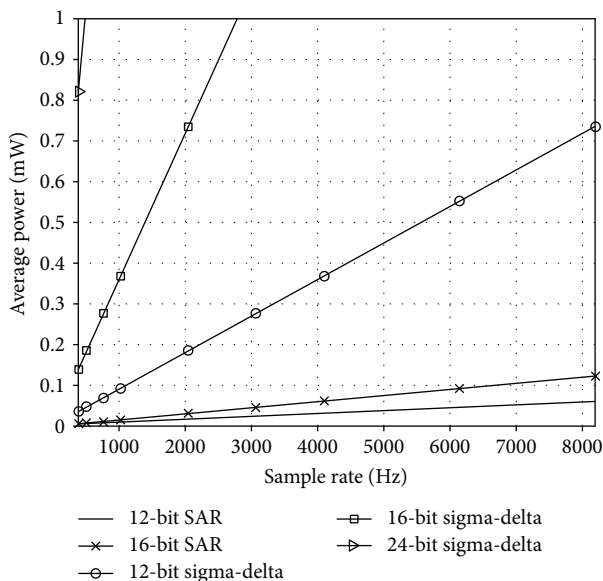| Technology	Component	ENOB	Nominal sample rate	Average power at 1 kSPS
12-bit SAR	ADS7924 [49]	11	10 kS/s	10 μW
16-bit SAR	AD7684 [50]	14	100 kS/s	15 μW
12-bit sigma-delta	ADS1014 [51]	12	3.3 kS/s	92 μW
16-bit sigma-delta	ADS1114 [52]	16	0.860 kS/s	368 μW
24-bit sigma-delta	ADS1251 [53]	19	20 kS/s	1.95 mW

FIGURE 2: Comparison of average power consumption of 12-, 16-, and 24-bit SAR and sigma-delta ADCs operating in duty cycle mode with respect to sample rates from 250 Hz to 8 kHz.

components supporting "burst mode," "single-shot," or "auto-power-down" combined with fast wake-up times were selected for the analysis. The components of different nominal throughput were compared to evaluate the potential benefit of using short active and long-sleep periods.

A list of evaluated components alongside with their respective performance is listed in Table 2. Each component's total active time was estimated from its power-up time, number of erroneous conversion samples, and conversion time. Finally, we extrapolated average power spent by each component at sampling frequencies spanning from 250 Hz to 8 kHz. Average power was obtained from its total active time, power declared at their nominal sampling rate, and the declared sleep mode power.

Results shown in Figure 2 confirm that the lowest consumption is obtained for ADCs featuring a combination of the lowest supply voltage and the lowest active and sleep current and supporting high throughput (i.e., short active time). Specifically, for the required range of sample rates, SAR models show clear advantage in the average power

consumption over the sigma-delta counterparts. In case of 12-bit converters, SAR yields around 9 times lower average power than the comparable sigma-delta model, while at 16 bits, the difference increases to more than 24 times. With the 24-bit sigma-delta ADC, 1 mW suffices for the average sample rate of merely 500 Hz. Thus, 16-bit SAR ADC architecture is considered optimal for respiratory sound digitization. It consumes in range of 4 to 123 μW while operating at sample rates between 250 Hz and 8 kHz. In respect to 12-bit SARs, 16-bit SAR requires about 50% more power.

2.3. Signal Processing

2.3.1. Processing Cores for Respiratory Sound Classification.
Power consumption required for execution of the wheeze detection was evaluated for two studied algorithms: for empirical rule-based frequency-tracking algorithm [12] and the HMM-based wheezing frequency algorithm [22, 23]. Both algorithms were first implemented, trained, and tested in Matlab on a dataset of prerecorded respiratory sounds. Details on dataset's size, constitution, and origin are given in [12, 23]. After that, algorithms' performance and execution speed were individually verified on a selection of representative processing cores: an audio DSP TMS320C5505 (Texas Instruments, [12]), an ARM Cortex-M4 Bluetooth 4 SoC BGM113 (Silicon Labs), and an ARM Cortex-A9 smartphone SoC Exynos 4 (Samsung Galaxy S2, [22, 23]). In the following section, we generalize these findings, by assessing algorithms' execution efficiencies for a broad range of commercial processing core architectures.

As a universal rule, we focused on processing cores featuring the lowest active-state power at the highest operating frequency, in combination with the low sleep state power, due to their potential to yield the lowest average power [54, 55]. Table 3 summarizes a list of the cores used in the benchmark. Three categories of processing cores were analyzed. The first was audio DSP processors. Advantage of dedicated DSPs is that their proprietary cores are designed to feature the architectural features accelerating execution of the DSP functions, such as multiply-and-accumulate units, barrel shifters for floating point operations, vector multiply, hardware FFT coprocessors, specific data transport I/O units such as I^2S [56]. Also, they are typically supported with extensive library of software functions for the audio processing. Lowest-power 16-bit fixed-point DSP cores were evaluated for onboard signal processing: TMS320C5535 (Texas Instruments) and ADSP2188N (Analog Devices). They were compared to a legacy 16-bit 56xxx core MC56F8006 (Freescale Semiconductors) and higher powered 32-bit ColdFire core MCF51MM128 (Freescale Semiconductors).

The second category was general purpose MCUs. Here, we compared general purpose, high-performance 32-bit ARM Cortex-M3 (STM32L151C8, ST Microelectronics) to lower-powered ARM Cortex-M0 (LPC1102, Linear Technologies). Also, 32-bit ARM cores were evaluated against proprietary 16-bit MSP430 core-executing code from an ultralow power ferroelectric (FRAM) program memory (MSP430FR572x, Texas Instruments). Also, a dedicated signal acquisition controller based on an older 16-bit ARM 7

TABLE 3: Parameters of the tested digital signal processors.

Processing core	Component	Freq., MHz	Active, μW/MHz	Sleep, μW
Audio DSP, 32-bit ColdFire	MCF51MM128 [57]	50	1740	84.0
Audio DSP, 16-bit 56800E	MC56F8006 [58]	32	4282	521.4
Audio DSP, 16-bit ADSP-21xx	ADSP2188N [59]	80	562	180.0
Audio DSP, 16-bit C55xx	TMS320C5535 [56]	100	220	220.0
High-perf. MCU, 32-bit ARM Cortex-M3	STM32L151C8 [60]	32	540	25.0
Signal acq. MCU, 16-bit ARM 7 TDMI	ADUC7060 [61]	10	775	137.5
Low-power MCU, 32-bit ARM Cortex-M0	LPC1102 [62]	50	462	6.6
Low-power FRAM MCU, 16-bit MSP430	MSP430FR572x [63]	24	275	19.2
Bluetooth SoC, 8-bit 8051	CC2541 [64]	32	628	2.7
Bluetooth SoC, 16-bit ARM Cortex-M0	nRF51422 [65]	32	495	6.9
Bluetooth SoC, 32-bit ARM Cortex-M4	BGM113 [66]	38.4	307	2.7
Bluetooth SoC, 32-bit ARM Cortex-M3	CC2640 [67]	48	110	4.9
Bluetooth SoC, 32-bit ARM Cortex-M4	nRF52832 [68]	64	100	1.7

core coupled with a high-resolution ADC (ADUC7060, Analog Devices) was included in the test.

The third category was processing cores embedded in system-on-chip (SoC) Bluetooth 4 communication modules. The latest generation SoC featuring 32-bit ARM Cortex-M3 (CC2640, Texas Instruments) is compared to two different ARM Cortex-M4 cores (nRF52832, Nordic Semiconductors; BGM113, Silicon Labs). Aforementioned cores were compared to previous generation SoCs featuring lower-powered processing cores: 16-bit ARM Cortex M0 (nRF51422, Nordic Semiconductors) and 8-bit 8051 core (CC2541, Texas Instruments).

Each core was benchmarked with the respiratory sound classification algorithms [12, 23]. Number of mathematical operations constituting each algorithm was modelled using analytical expressions derived in [12, 23], respectively; Number of additions and the multiplications was modelled separately as elementary instruction types with different power cost. Respective execution models show that the execution times of both test algorithms are dominantly dependent on two factors: (1) number of the observed frequency states (bins) M defined by DFT frequency resolution and (2) the spectral content of the input signal. Namely, execution of both algorithms is linearly proportional to the number of frequency lines L. Thus, in all test cases, M and L are kept identical for both algorithms to enable the comparison. M was varied in a span ranging from 32 to maximally 128 frequency states. For input signal containing wheezing, probabilities of occurrence of monocomponent ($L = 1$), two-component ($L = 2$), and three-component ($L = 3$) wheezing are uniformly distributed.

Motivated by the execution time's dependency of the signal content, a test environment was constructed to assess the dependency of the average processing power consumption with respect to the symptom severity, simulating realistic operating conditions. Symptom severity was modelled by two variables: (1) wheeze rate, and (2) symptom rate. Wheeze rate, a percentage of respiratory cycle obstructed by wheezing, was varied in the range of 0% to 50% (e.g., 25% wheeze

rate is reported as a severe obstruction of airways, experienced only during asthmatic attacks [7, 8, 69, 70]). Symptom rate is the frequency of occurrence of respiratory cycles containing wheezing, expressed as percentage, ranging between 0 and 100%.

For each combination of wheeze rate, symptom rate, and number of processed frequency states (M), execution time of each algorithm was calculated on each processing core. Cores' operating frequency, register width with respect to assumed 16-bit data width, and cost of multiplication with respect to addition were taken into consideration. Knowing the intervals between the consecutive processing tasks, processing duty cycle was calculated from the execution time (i.e., portion of time spent in active state). Finally, the average processing power consumption was calculated, using the power spent in active and sleep state. As processors generally feature more than one low-power state, sleep power was estimated to correspond to the low-power state, in which all the required periphery for short wake-up and periodic sampling by ADC (DMA and PLL) is operative, and the memory content is retained.

Example results in Figure 3 show an increase in average power and processing duty cycle proportionally to the symptom rate and the wheeze rate, for the HMM-based algorithm. The figure contrasts the algorithm execution on two different processing cores: 100 MHz 16-bit audio processing DSP (C5535, Texas Instruments) and 25 MHz 16-bit MSP430 general purpose low-power MCU. Due to 4 times lower clock frequency, processing on the MCU results in proportionally higher processing duty cycle (worst-case approx. 22% active time, with respect to 5% on the DSP). However, due to lower sleep power of the MCU, comparable power consumption is achieved by both cores, with the audio DSP being approximately only 7% more efficient.

In a similar manner, average power consumption distributions of Bluetooth 4 SoCs featuring different processing cores are compared in Figure 4. The best overall results are obtained on the 64 MHz ARM Cortex-M4 core (nRF52832), ranging from 308 to 452 μW. 48 MHz ARM

(a) Power consumption, 100 MHz audio DSP

(b) Active-state duty cycle, 100 MHz audio DSP

(c) Power consumption, 25 MHz 16-bit MCU
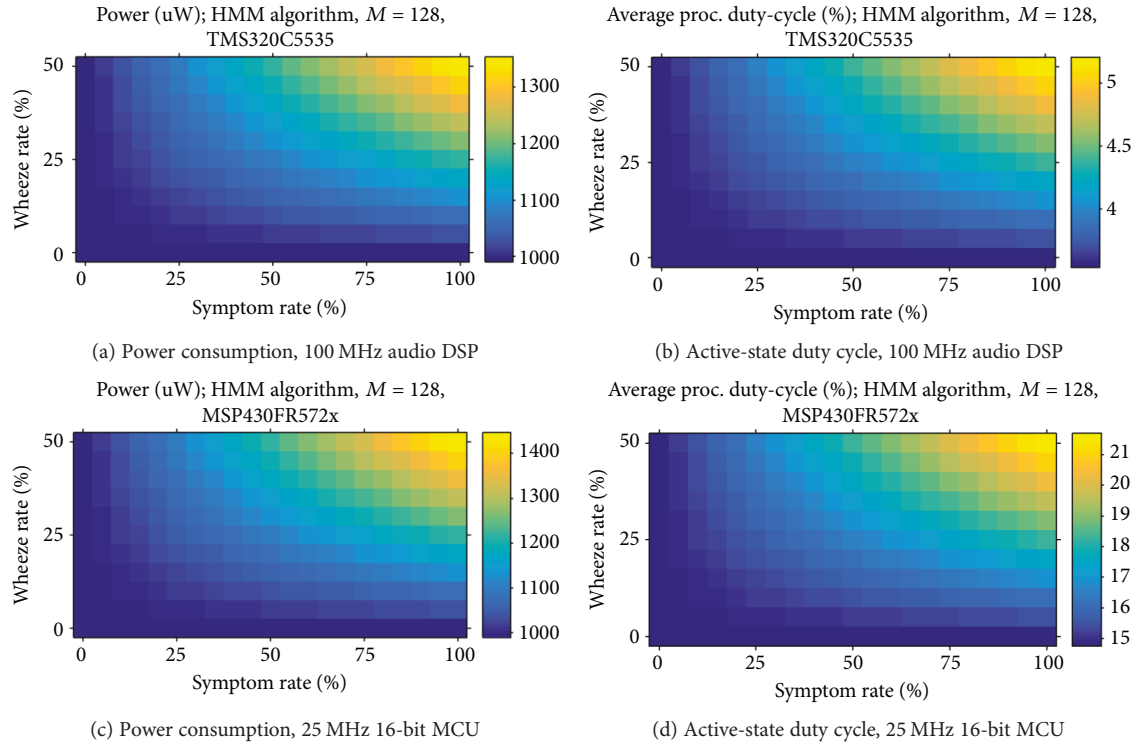
(d) Active-state duty cycle, 25 MHz 16-bit MCU

FIGURE 3: Comparison of processing power consumption and active-state duty cycles of a typical 16-bit audio DSP and general purpose low-power MCU.

(a) Power consumption, 64 MHz ARM Cortex-M4

(b) Power consumption, 48 MHz ARM Cortex-M3

(c) Power consumption, 38 MHz ARM Cortex-M4
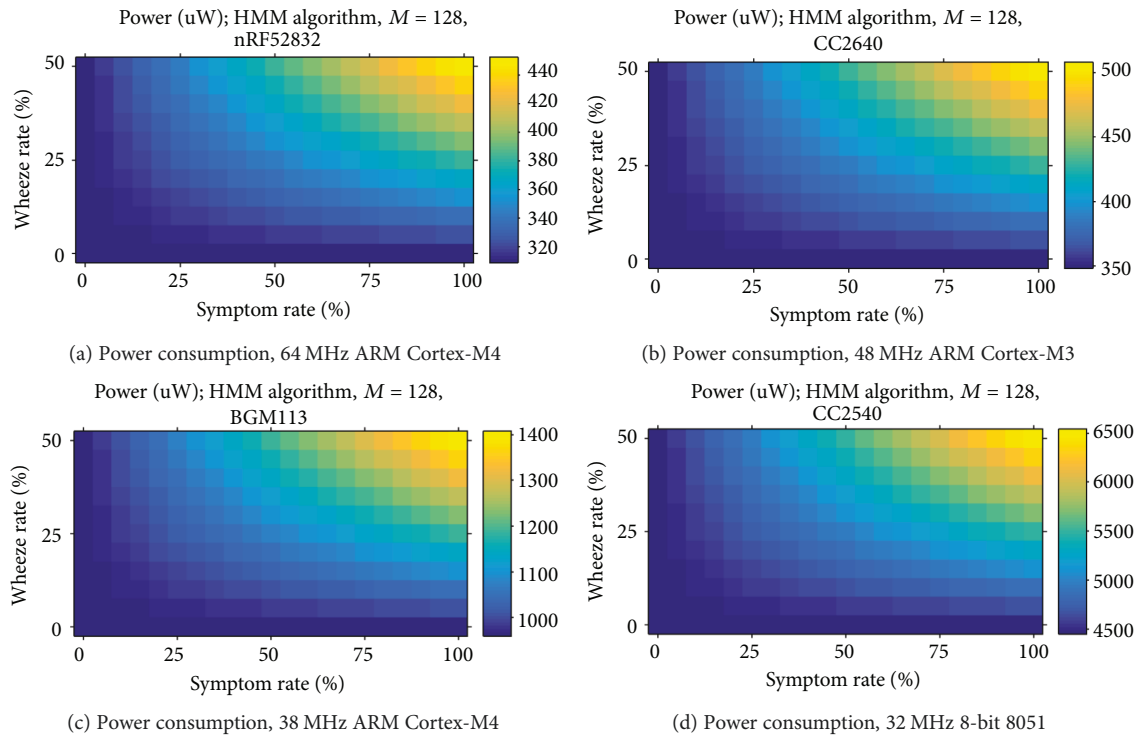
(d) Power consumption, 32 MHz 8-bit 8051

FIGURE 4: Comparison of processing power consumption on four Bluetooth 4.x SoCs.

Cortex-M3 (CC2640) lags behind as the second, consuming 348 to 505 μW. BGM113, although featuring the same core as the nRF52832 (ARM Cortex M4), yields 3 times higher power consumption, mainly due to 3 times higher active power and approximately 2 times lower clock frequency. Legacy Bluetooth SoC module featuring 8051 (CC2541) proves suboptimal for respiratory sound processing, due to 8-bit architecture, low clock frequency, and so on.

(a) Ranking by average-case power consumption

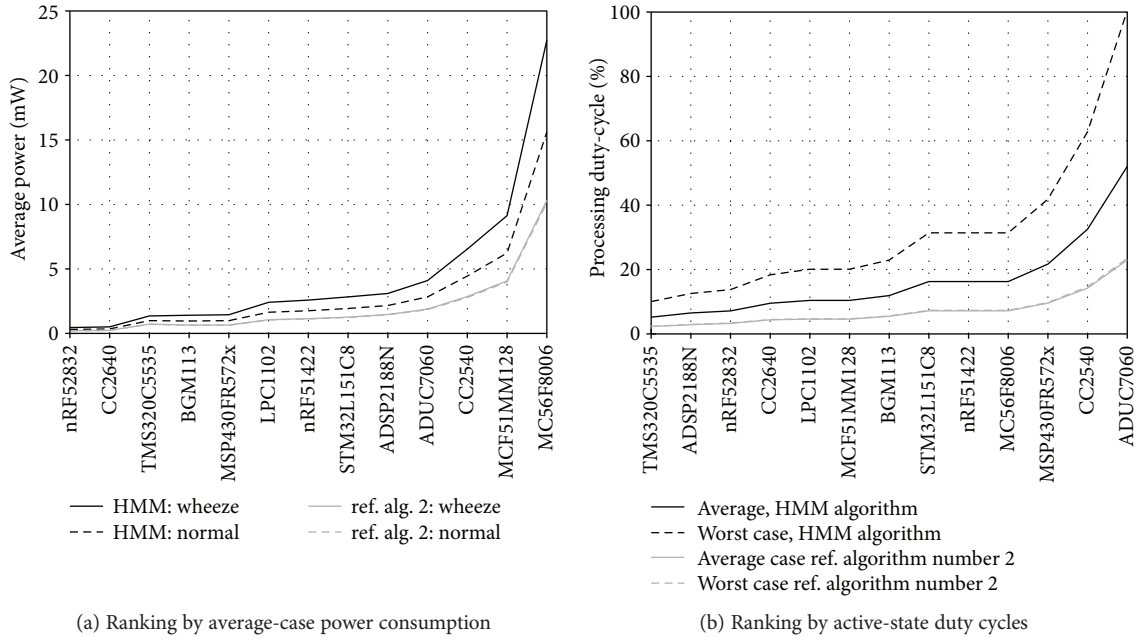(b) Ranking by active-state duty cycles

Figure 5: Wheeze classification algorithms' benchmark performed on a list of processing cores.

The overall results are shown in Figure 5. In Figure 5(a), all cores are sorted by the worst-case power consumed for processing of wheezing with respect to the power consumed for processing of normal respiratory sound, by both algorithms. The results are shown for fixed, maximal $M = 128$. It can be seen that for HMM-based algorithm, the processing of wheezing may require approximately up to 45% more power than processing of normal respiratory sounds. On the other hand, referent algorithm shows negligible difference, as it gradually reduces dimensionality of data.

The best results are obtained for ARM Cortex-M4 and M3 cores embedded in Bluetooth 4 SoCs (nRF52832 and CC2640) and with dedicated low-power audio C55xx DSP (TMS320C5535). On the other hand, the worst efficiency is obtained with ADUC7060 signal acquisition controller, high-performance 32-bit ColdFire audio DSP core (MCF51-MM128), and the legacy 16-bit 568xx DSP core (MC56F8006).

The average-case and the worst-case processing duty cycles are compared in Figure 5(b), to potentially identify cores failing to meet constraints of the real-time processing. The results show that least resources are spent by dedicated audio DSPs TMS320C5535 (worst case 10% of processing time) and ADSP2188N. On the other hand, it is shown that due to low maximal clock frequency (only 10 MHz), ADUC7060 signal acquisition controller hardly meets the worst-case real-time requirements when running the HMM-based classification algorithm (with $M = 128$ hidden states). Also, 8051 and MSP430 spend high portions of their processing time, 60% and 40%, respectively.

Finally, efficiencies of the HMM-based and referent crest-tracking respiratory sound classification algorithms are compared with respect to the number of the processed frequency states (bins) M in Figure 6. At the maximal $M = 128$, the

HMM-based algorithm requires approximately 1.5 times more power with respect to the referent algorithm. However, as execution time of the HMM-based algorithm scales with M^2, its power requirements may be significantly reduced by lowering the number of processing states (e.g., focusing the algorithm on a narrower frequency band of interest). The red line depicts the intersection at which processing power consumption requirements of both algorithms are equalized. In worse case of the processing of respiratory sounds with wheezing, this boundary lays at approximately $M = 64$. For HMM algorithm, operating on signal sampled at 2 kHz, this would mean narrowing the bandwidth of the observed signal to 500 Hz. Nevertheless, this is considered sufficient for classification (see [12]).

2.3.2. Processing Cores for CS Encoding. Here, we evaluate power requirements for the CS signal encoding implemented by sub-Nyquist sampling of the analog input signal at the nonuniform pseudorandomly spaced sampling instants [30, 31]. The choice of encoder was motivated by the fact that the mentioned CS encoder design requires minimal number of digitized signal samples, thus enabling the highest savings of active power in signal acquisition subsystem (i.e., ADC's and MCU's active time spent on acquisition, data handling, and storage) [71]. Also, it enables for simple implementation in microcontroller's firmware or a dedicated digital hardware [31]. Finally, it enables for easy and energy-efficient synchronization of pseudorandom number generators between sensor and the receiver/smartphone side (with minimal data transmission overhead) [30]. As the ADC power has already been covered in Section 2.2, this analysis focuses on power spent on the processing tasks implementing the LFSR pseudorandom number generator and the sampling period scheduler blocks in the MCU software and operation of the 16-bit timer MCU peripheral unit.

(a) Processing sounds with wheezing
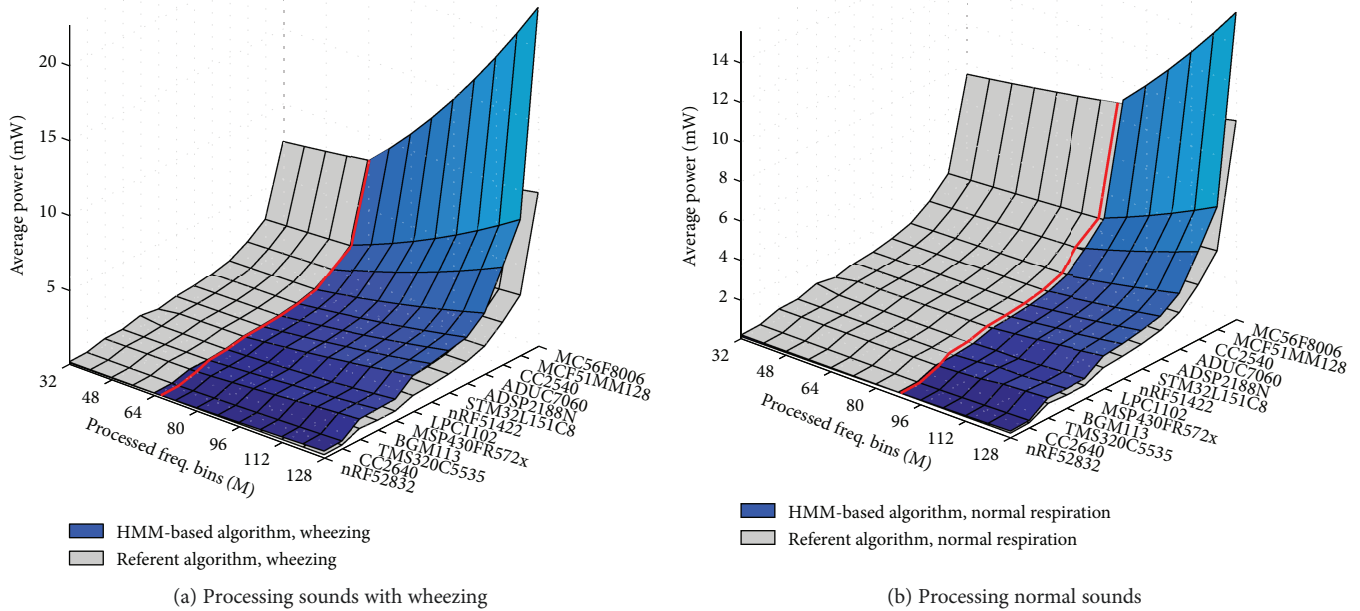
(b) Processing normal sounds

Figure 6: Power required for HMM-based and referent crest-tracking respiratory sound classification algorithms with respect to number of processed frequency states (bins) M.

The processing tasks required for sampling of a single CS-encoded sample include (1) generation of the next LFSR pseudorandom output, (2) scheduling of the next sampling instant, (3) setup of the timer, (4) and triggering the ADC conversion. Cost of the MCU implementation within Free-RTOS was empirically verified on a prototype implementation on MSP430 to approximately 150 instructions per single CS-encoded sample. Cost of the timer operation was estimated from the MCU manufacturer data (e.g., [67]) and scaled with the clock frequency.

The total cost of CS encoding was simulated for a range of CS sub-Nyquist sample rates corresponding to compression ratios of 2x, 4x, 5.33x, and 8x with respect to the Nyquist sampling frequency of 2 kHz. In addition to MSP430, power cost of the CS encoding was simulated on several additional MCU cores: ADUC7060 signal acquisition controller and MCUs within Bluetooth 4 SoCs: nRF52832, CC2640, BGM113, nRF51422, and CC2540.

Average power and the processing duty cycle required for the CS encoding with the increase of compression ratio are shown in Figure 7. The results show that the most efficient implementation may be achieved on Bluetooth 4 SoC ARM Cortex cores. On the nRF52832 power for the CS encoding ranging from 17 to 63 μW, while on the CC2640, it costs 22 to 73 μW. It is shown that CS encoding on the nRF52832 in the worst-simulated case spends less than 1% of the processing time.

2.4. Bluetooth Communication. Bluetooth 4 (i.e., Bluetooth Smart, Bluetooth Low Energy) radio technology was evaluated for the wireless data transfer, as it enables for interoperability with smartphones and medical certification [72],

while retaining low-power operation. The highest level of integration of Bluetooth radios is provided with system-on-chip (SoC) modules, packaging together a digital radio, a radio controller implementing the Bluetooth stack, an application processor, and a variety of standardized embedded peripheral interfaces.

The following state-of-the-art Bluetooth 4 SoCs were analyzed: CC2640, CC2541 (Texas Instruments), BGM113 (Silicon Labs), and nRF52832 (Nordic Semiconductors). Table 4 compares their average power in the most typical operating states: radio transmission (TX), radio listening (receiving, RX), and sleep. The power reduction in TX and RX in order of 2 to 3 times can be observed when comparing the previous and the actual generations of Bluetooth 4 SoCs (CC2541 with respect to CC2640, nRF52832, and BGM113). The increase of processing capabilities enables for the implementation of respiratory sound classification algorithms (e.g., [12, 13, 16]) onboard SoC's application processor.

Bluetooth 4 communication protocol is designed with the premise of fostering the lowest average power, by featuring intermittent, short active time (TX, RX) of the radio, in combination with long sleep time in between connection intervals. Data packets are exchanged only at predefined periodical *connection intervals*, during the so-called *connection events*. Upon completion of the connection event, the radio is put to sleep until the next one [73]. Duration of a connection event is minimized by high communication throughput (typically 2 Mbit over the air). Typical waveform of the CC2640 radio's power supply current measured during the connection event, segmented into a sequence of common power states, is shown in Figure 8 (see descriptions for the labels 1 to 6).

(a) Power consumption for CS encoding

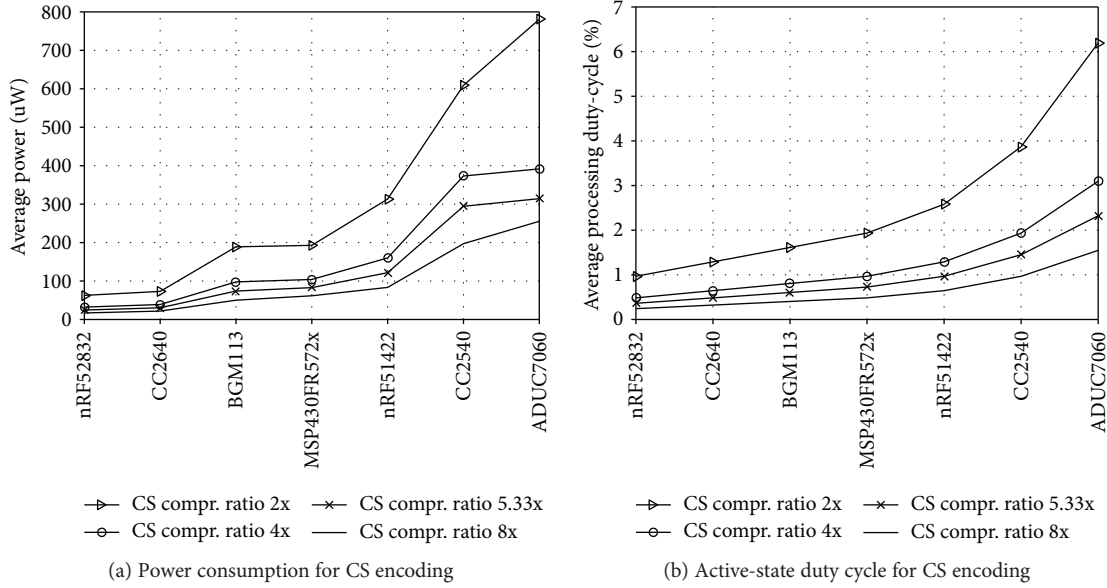(b) Active-state duty cycle for CS encoding

Figure 7: Comparison of processing cores' consumption and active-state duty cycles for CS encoding.

Table 4: Parameters of the tested Bluetooth 4 SoC modules.

Component	Application processor	TX, mW	RX, mW	Sleep, μW
CC2541 [64]	8-bit 8051	36.4	35.8	2.0
BGM113 [66]	32-bit ARM Cortex-M4	16.3	16.1	2.6
nRF52832 [68]	32-bit ARM Cortex-M4	13.1	12.0	2.8
CC2640 [67]	32-bit ARM Cortex-M3	11.3	10.9	1.9

Due to numerous parameters influencing durations of each power state, we focused our power analysis on the CC2640, as it offers extensive Bluetooth power estimation guidelines [73], power simulation tools [74], and data [67]. The analysis assumes following parameters and limitations of CC2640. The power was measured at the supply voltage of 1.8 V. Output power of the transmitter was set to 0 dBm as communication between the sensor and the smartphone is taking place at the very short range (i.e., <10 m). Maximal payload size during a single connection event was limited by the Bluetooth software stack to maximally 256 bytes. Time between the successive connection intervals may range from 7.5 ms to maximally 4.0 s.

With the given constraints, the average power consumption spent on communication was calculated for each of the three operating scenarios: Firstly, in the scenario of streaming of uncompressed data, we analyzed cases of sampling (i.e., streaming) rates of 8 and 2 kHz. There, 8 kHz corresponds to the case of using the referent empirical rule-based spectral crest-tracking classification algorithm [12] on the smartphone (requiring DFT block size $N = 512$, with 75% overlap). On the other hand, 2 kHz corresponds to classification on smartphones by the HMM-based algorithm

[21, 22], on signal blocks of $N = 256$ samples, overlapped by 75% as well.

Secondly, in the scenario of streaming of the CS-compressed signal, we analyzed 4 different compression ratios with respect to the 2 kHz Nyquist frequency: 2x, 4x, 5.33x, and 8x. Payload size is calculated with respect to original signal block size of $N = 256$ and 75% overlap, as the HMM-based classification algorithm [21, 22] is assumed. Also, each TX payload size is increased by 2 additional bytes needed for the pseudorandom seed.

Finally, in the third scenario of the respiratory sound classification onboard sensor, the binary block-wise classification outcome is encoded in a periodically sent report messages. The connection period and payload size depend on the classification algorithm (sampling frequency, signal block size) and payload content, whether the stream of raw binary classification outcomes corresponding to each signal blocks is sent or if wheeze rate is calculated for a predefined temporal window. For all scenarios, 2-byte acknowledged RX message is assumed.

In the analysis, relative contributions of the payload size and connection interval were compared for each operating scenario, by accumulating (buffering, storing) the TX data on the sensor across multiple connection intervals until reaching the maximal payload size and then transmitting it in bulk. Table 5 summarizes the tested scenarios, nominal payload sizes, and the span of possible connection intervals supporting the transmission, given the payload size limitations (i.e., 256 bytes on CC2640).

Spans of the average power required for the communication in each operating scenario with respect to time between successive Bluetooth connection events are shown in Figure 9. It is shown that due to very short active times (i.e., high data rate), sleep power spent in between the connection intervals influences the average communication power much more than the change of payload size. This
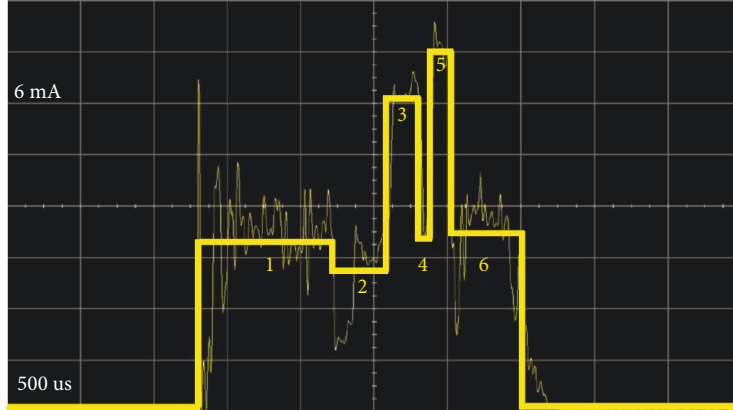
FIGURE 8: Example waveform of current measured during a single Bluetooth 4 connection event. 1—real-time operating system wake-up, radio setup; 2—radio turned on, transition to RX; 3—radio receiver listening (RX); 4—transition from RX to TX; 5—radio transmission (TX) of the packet; 6—Bluetooth stack processing the received packets, setup of sleep timer, going to sleep [73].

TABLE 5: A list of tested communication scenarios, with best-case average power.

Scenario, case	Min. payload TX/RX, bytes	Conn. intvl. span, ms	Min. avrg. power, μW
Nyquist rate streaming, f_s 8 kHz	256/2	16–32	914
Nyquist rate streaming, f_s 2 kHz	128/2	31.25–62.5	373
CS streaming, compr. 2x	(64+ 2)/2	31.25–125	168
CS streaming, compr. 4x	(32+ 2)/2	31.25–250	81
CS streaming, compr. 5.33x	(24+ 2)/2	31.25–250	79
CS streaming, compr. 8x	(16+ 2)/2	31.25–250	77
Class. result reporting	2/2	125–4000	8

causes the average power to exponentially fall when increasing the connection interval. Thus, we propose to maximally prolong the (sleep) time between connection intervals by accumulating the data, filling each transmission packet up to the maximal payload size.

The best-case results for each scenario obtained in a similar manner are highlighted in Table 5. The Nyquist rate data streaming proves most costly, costing $914\,\mu$W at 8 kHz. Drastic decrease of power in case of the Nyquist rate streaming at 2 kHz originates from doubling the connection interval. The identical mechanism is the reason for decrease from 168 to 81 μW when stepping-up from the CS compression ratio of 2x to 4x. On the other hand, minimal difference in the average power is observed in cases of identical connection intervals, where only the payload size was increased (e.g., at the CS compression ratios 4x, 5.33x, and 8x). In the scenario of onboard classification, the minimum average power of only 8 μW is achieved at the maximal connection interval of 4 s.

3. Wearable Sensor's Total Power Consumption

Analysis of the total sensor power consumption is given here. Three operating scenarios from Section 1 are compared: (1) the onboard classification, (2) the CS signal streaming, and (3) the streaming of uncompressed Nyquist rate signal. For each scenario, the total power is calculated by summing up contributions of the acoustic sensor, analog conditioning, A/D conversion, processing, and Bluetooth communication subsystem, based on results presented in Section 2.

To enable the comparison of all three operating scenarios, the analysis assumes a wearable sensor constituting of common subsystem components. An analog MEMS microphone (such as ICS-40310) is combined with analog front-end from Section 2.1 for conditioning and 16-bit SAR ADC (e.g., AD7684) for digitalization. Processing and communication are implemented on the Bluetooth 4 SoC featuring the ARM Cortex-M4 processing core, proven optimal for both onboard classification and CS-encoding tasks (see Section 2). For completeness, the analysis is based on the representative CC2640 SoC. In the scenario of onboard classification, the number of processed frequency states is equalized for both classification algorithms to $M = 64$ to yield comparable power. In the CS streaming scenario, range of the CS compression rates was chosen to yield comparable classification performance with respect to Nyquist rate signal acquisition. According to [21, 30], the classification accuracy is degraded by less than 5% for the CS compression ratios lower than 5.33. Thus, the power analysis focused on compression ratios spanning from 2x to 5.33x.

The total power consumption of the CC2640-based sensor is compared in Figure 10. Being constantly powered and architecturally identical in all scenarios, the microphone and the analog conditioning circuit contribute equally to the total power with 101 μW. Power contributions of the remaining subsystems are scenario dependent.

In the scenario of onboard classification, lower total power is obtained for the case of the HMM-based algorithm operating on a signal sampled at 2 kHz. The total average power was 320 μW, with a major share of 56% being taken
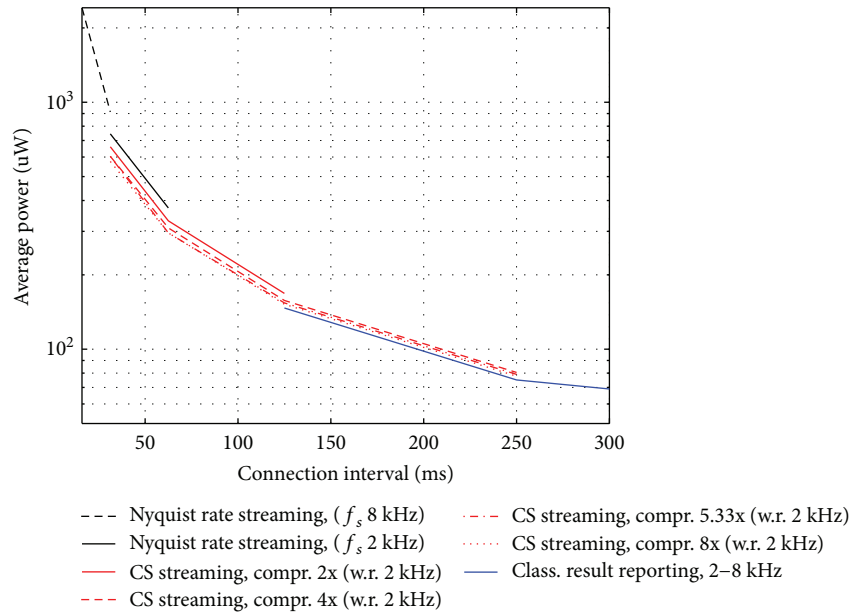
FIGURE 9: Average power spent on Bluetooth 4 communication with respect to time between successive connection events.
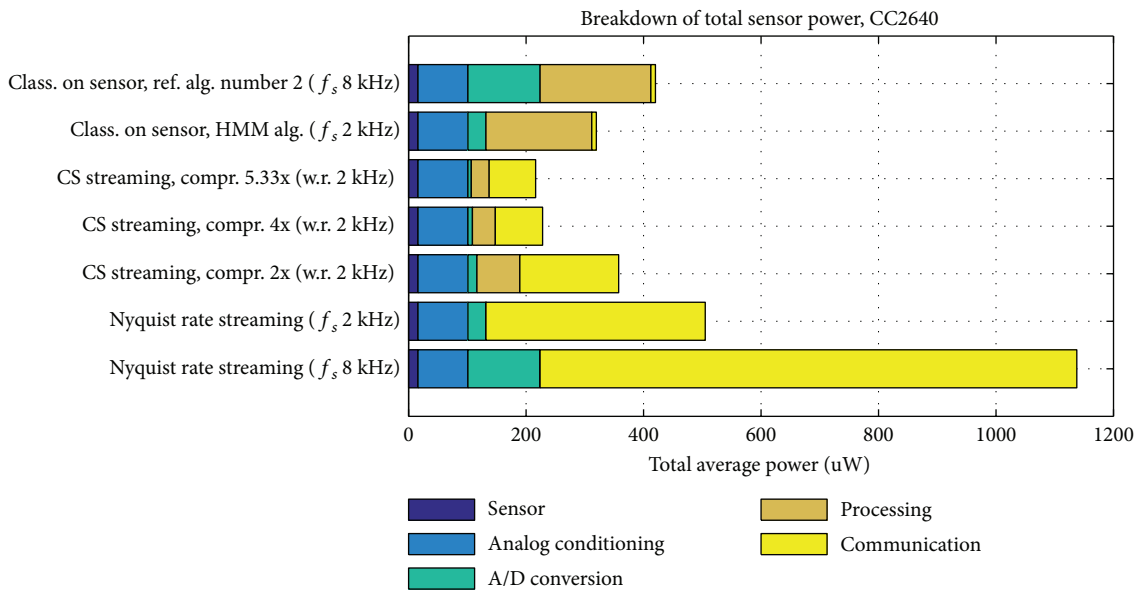

FIGURE 10: Comparison of total power consumptions for different operating scenarios. Breakdown of power consumption per subsystems.

by the classification algorithm. As a comparison, the classification using the referent crest-tracking algorithm, requiring the signal digitized at 8 kHz, results in 31% higher sensor power (i.e., 420 μW).

In the scenario of streaming of the CS-encoded signal, total power scales down as expected when increasing the compression ratio. At higher compression ratios of 4x and 5.33x, it yields 228 and 216 μW, respectively. Majority of this power is spent on communication. However, the significant processing share related to CS encoding occurring at low compression ratios (e.g., 76 μW for compr. ratio of 2x) points to inefficiency of the MCU software implementation of CS encoding.

As the scenario of uncompressed signal streaming virtually excludes any processing, the largest portion of the power is spent on communication (proportional to the sample rate). In the best case (at 2 kHz, assuming classification using the HMM-based algorithm on smartphone), the total power was 505 μW. At 8 kHz, it doubled to 1138 μW. Thus, streaming the uncompressed signal proves to be the worst solution.

Power analysis confirmed that the sensor performing the CS encoding, operating at the compression rate higher or equal than 4x, yields the lowest total power. By requiring less power for processing, it outperforms the best case of onboard classification 1.8 times (HMM algorithm with sample rate at
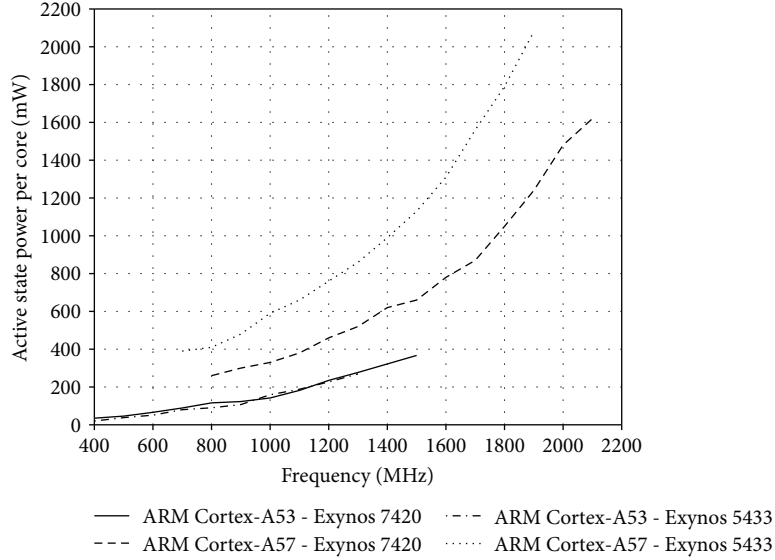
FIGURE 11: Range of operating frequencies and per-core active-state power consumption of ARM Cortex-A53 and A57 cores. Data were taken from measurements [75, 76].

TABLE 6: Analytical models of computational and storage complexities of the OMP algorithm per single iteration, taken from [35].

Implementation	Computational cost	Storage cost
OMP, QR factorization	$2MK + M + A + N$	$2(M + 1)K + 0.5\,K(K + 1) + A + N$
OMP, Cholesky factorization	$3A + 3\,K^2 + 2\,M + N$	$0.5\,K(i + 1) + A + M + 2\,K + N$

2 kHz). Also, by reducing the communication cost, it yields 2.2 times lower total power with respect to uncompressed signal streaming (at 2 kHz). This confirms usability of the practically implemented CS encoding in systems where off-loading the sensor in terms of power consumption is the primary design criterion.

4. Smartphone Power Consumption

In the smartphone-centric wheeze quantification sensor system (i.e., in both signal streaming scenarios), the major contributors to smartphone's power consumption are communication and signal processing. Processing includes respiratory sound classification and additionally, in the scenario of CS streaming, CS reconstruction.

Processing cost is evaluated for typical system-on-chips (SoC) integrating multiple processing cores, radio, and peripheral units. Such are the Samsung Exynos 7420 SoC [75] (Samsung Galaxy S6, Galaxy Note 5) and Samsung Exynos 5433 SoC [76] (used in Samsung Galaxy Note 4, Galaxy Tab S2). Both SoCs encapsulate 8 cores in the so-called big.LITTLE configuration, consisting of a cluster of 4 low-power ARM Cortex-A53 cores [77] and 4 high-performance ARM Cortex-A57 [78] cores. Both cores are based on ARMv8-A 32/64-bit architecture and feature architectural extensions supporting efficient digital signal processing (i.e., NEON SIMD, VFPv4 floating point unit). Figure 11 compares Samsung Exynos 7420 and 5433 SoC's per-core active-state power consumptions.

4.1. Power Cost of CS Reconstruction on Smartphone. Power cost of the CS reconstruction on smartphones was analyzed for the representative orthogonal matching pursuit (OMP) algorithm [30, 32, 79]. The OMP algorithm estimates a K-sparse solution to the underdetermined $M \times N$ system ($M < N$), by organizing it into the iterative procedure, in whose each iteration one-after-another solution vector element is estimated by solving an overdetermined system (of the dimensionality growing with each iteration) using the linear least squares method. As the calculation of the least squares presents a major bottleneck of the algorithm, we analyzed two approaches to solving the least squares, one by utilizing QR factorization and the other featuring the Cholesky factorization. The computational complexities of both algorithm implementations were modelled analytically, based on the expressions derived for per-iteration computational cost [35], shown in Table 6. N stands for the original signal length, M is the compressed signal length, K is the number of (sparse) signal components, i is the iteration number, and A is the calculation cost of the CS measurement matrix. The total cost was evaluated for the number of iteration (termination condition) set to exactly K reconstructed frequency components.

All simulations were performed using the fixed original DFT/DCT block size of $N = 256$ ($f_s = 2$ kHz), with the number of iterations K parametrized in range 4–32 and the number of subsampled signal elements M in the range 128–32, yielding the compression ratios N/M of 2–8. The model was fed with the architectural parameters of ARM Cortex-A53 and ARM Cortex-A57 processing cores featured in Exynos
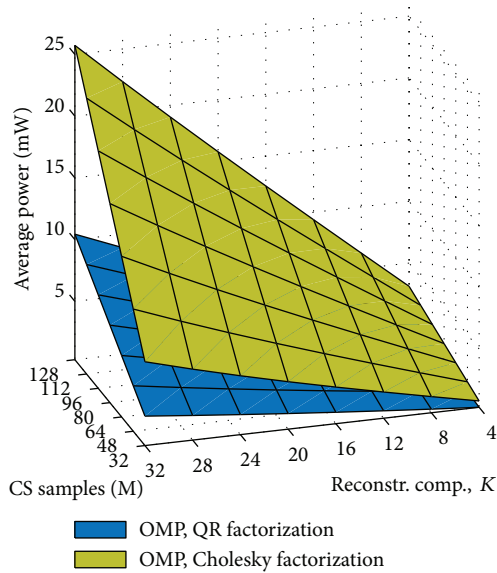
FIGURE 12: Average power required for OMP execution on smartphones. Comparison of implementations featuring QR and Cholesky factorization. $N = 256$. Single ARM Cortex-A53 core running at 1.5 GHz, embedded in Exynos 7420 SoC.

7420 smartphone SoC. Single-core operation is simulated. From estimated total processing times and required processing intervals, occupancies of the processing cores (i.e., processing duty cycles, in %) and the associated increment in cores' average power consumption were calculated. Increments in average power consumptions obtained by executing the OMP featuring QR and Cholesky factorization are compared in Figure 12 for case of the smartphone ARM Cortex-A53 core running at 1.5 GHz. In general, OMP's processing time (i.e., average power) grows proportionally with both K and M and is more sensitive to M. The implementation featuring QR factorization yields about 2.5 times less power than the implementation with Cholesky; thus, the version featuring QR is considered optimal and is further analyzed within the rest of the study.

Spans of average powers required for OMP execution on A57 and A53 processing cores within Exynos SoC with respect to range of cores' respective clock frequencies are shown in Figure 13(a). In the worst case, obtained for the maximal M and K, the power on the low-power A53 core (blue) ranges from 3.7 mW at 400 MHz to 10.4 mW at 1500 MHz. On the high-performance core A57 (orange), 13.8–32.9 mW is required, for the frequency span of 800–2100 MHz. Analogously, Figure 13(b) depicts associated processing time duty cycles. Processing time, dropping inversely with clock frequency, spans in the worst case from 2.8% to 10.7% on A53 (blue) and from 2.0% to 5.3% on A57 (orange).

However, in the typical use case, lower M is utilized in order to implement higher compression ratio. Also, depending on the signal sparsity (compressibility), the number of reconstructed spectral components K may be chosen reasonably low (e.g., $K = 8...16$). Thus, the power consumption would range from 1.2 to 2.4 mW on A53 core running at its maximal 1500 MHz. By scaling the frequency down to 400 MHz, the power drops to 400–850 μW, at the cost of

higher processing duty cycle, spanning between 1.2 and 2.4% of core's processing time. The analysis shows that single ARM Cortex-A53 processing core entails sufficient resources for real-time execution of the OMP (QR) CS reconstruction.

4.2. Power Cost of Respiratory Sound Classification on Smartphone. Resource requirements for respiratory sound classification were evaluated for case of the proposed HMM-based respiratory sound classification algorithm [21, 22] digitized at f_s of 2 kHz (DFT block size $N = 256$, 75% overlap). We analyzed the worst-case execution time obtained at the maximal symptom occurrence rate (100%) and for the wheeze rate of 50%. Number of the processed frequency states (i.e., DFT bins) M_{freq} and the processing core's frequency were taken as parameters. The processing core's parameters were set as for OMP in Section 4.1.

According to the results in Figure 14(a), the worst-case power cost of the classification on high-performance ARM Cortex-A57 core spans between 1.7 mW at 800 MHz and 4.0 mW at 2.1 GHz. On the other hand, lower power may be obtained on ARM Cortex-A53, ranging from 400 μW at 400 MHz to 1.2 mW at 1.5 GHz. The algorithm typically requires well below 1.0% of the processing time (see Figure 14(b)). In the case of the HMM-based algorithm optimized to operate on a reduced number of the frequency states (e.g., $M_{\text{freq}} = 64$), the average power on Cortex-A53, operating at the minimal clock frequency of 400 MHz, would drop to 158 μW, occupying around 0.45% processing time.

5. System-Level Power Consumption

The total system-level power consumption is comprised of the sensor's power analyzed in Section 3 and the power spent onboard the smartphone. Complexity of typical smartphone's hardware architecture (multicore processing units, multiple wireless communication interfaces, storage devices, screens, peripheral units, etc.), heterogeneity of hardware components, and high-complexity operating system software governing the allocation of hardware resources and power management policies complicate mW-accurate system-level modelling.

Total power of smartphone models featuring processing cores such as those analyzed in Section 4 typically ranges from 120 to 550 mW [31, 75]. In order to foster a meaningful comparison with the sensor's power, a simplified smartphone power-estimate simulating only the incremental contributions to the smartphone's active-state baseline power is given here. Estimates on smartphones' power requirements for processing (i.e., cost of CS spectrum reconstruction and respiratory sound classification) are based on results from Section 4. Also, it is assumed that the increment in smartphone's power consumption for Bluetooth communication is comparable to the power of the sensor (Section 2.4). Within these constraints, the total system-level power for each of three operating scenarios was estimated. Table 7 summarizes the findings.

The worst-case system-level power consumption is measured in the referent scenario featuring uncompressed data streaming, tested for the range of sampling/streaming

(a) Increment in core's average power
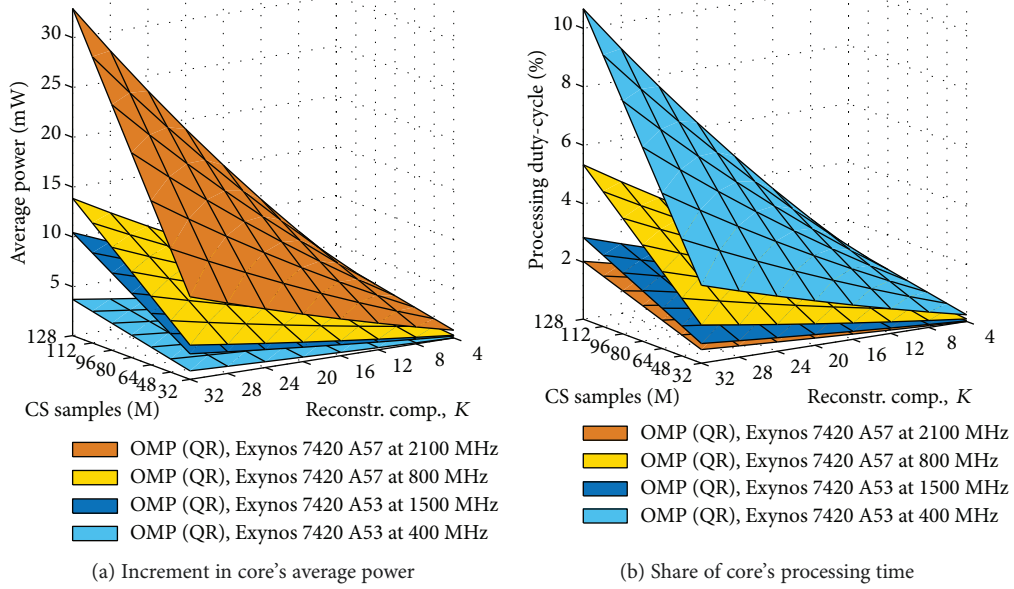
(b) Share of core's processing time

FIGURE 13: Span of processing resources required for execution of OMP on Exynos 7420 SoC ARM cores: low-power Cortex-A53 versus high-performance Cortex-A57.

(a) Increment in core's average power
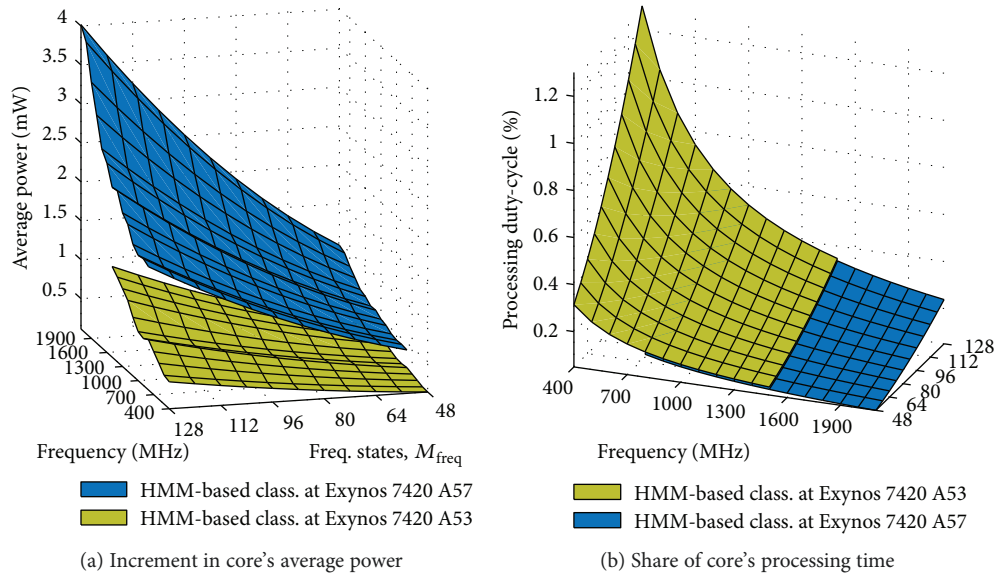
(b) Share of core's processing time

FIGURE 14: Span of processing resources required for HMM-based respiratory sound classification algorithm on Exynos 7420 SoC ARM cores (Cortex-A53, A57).

TABLE 7: Total system-level power consumption in the analyzed operating scenarios.

Scenario	Sensor, μW	Smartphone, proc., $\Delta\mu$W	Smartphone, comm., $\Delta\mu$W	Total, μW
Class. on the sensor, $f_s = 2$–8 kHz	320–420	—	8	328–428
CS signal streaming, $N/M = 5.33$–2	216–357	460–2080	79–168	775–2605
Uncompr. Streaming, $f_s = 2$–8 kHz	505–1138	158–632	373–913	1036–2683

rates f_s spanning from 2 to 8 kHz, and smartphone running HMM classification algorithm. At 2 kHz, the total power was estimated at 1.3 mW, spanning to 3.6 mW at sample rate of 8 kHz.

The minimum overall system-level power consumption is obtained in the scenario featuring classification onboard the sensor. Here, the smartphone is free of any signal processing tasks, and minimal power is spent on communication

between peer devices. Thus, the majority of system power is related to the sensor's power, keeping the total system's power below 428 μW.

In the CS signal streaming scenario, the total power ranges from 775 to 2605 μW, for respective CS compression ratios spanning from 5.33x to 2x. It features at least 2.4x higher total system-level consumption with respect to the sensor system with classification onboard the sensor but brings up to 25% savings in total system-level power with respect to the uncompressed signal streaming (at f_s of 2 kHz). Widespan of the total system-level power is primarily influenced by its dependency on the CS compression ratio (yielding higher power requirements at low compression ratio). Depending on the compression rate, cost of CS encoding on the sensor is 2.5x to 6x lower than the power required for execution of the HMM algorithm in the scenario of onboard classification. On the smartphone, CS reconstruction poses the largest bottleneck, costing 2.2 to 5.4 times more than classification with the HMM-based algorithm. The power cost of its implementation (estimated at 158 μW) is about 12% lower compared to the implementation on the wearable sensor (180 μW).

Power figures from Table 7 enable us to estimate the effective sensor system runtime. For example, a wearable sensor based around Texas Instruments CC2640 SoC, powered from a 200 mAh lithium coin-cell battery (CR2032) and a suitable DC/DC voltage regulator featuring 85% efficiency, would last for 50–66 days running onboard wheeze detection, depending on the wheeze detection algorithm. In the same conditions, a sensor node implementing CS sampling and streaming would operate for 59–98 days (depending on compression ratio), and the sensor node performing pure Nyquist rate sampling and streaming, only 18–42 days (depending on sample rate).

On the other hand, let us assume a smartphone consuming on average 125 mW, powered by a 2500 mAh Li-Ion accumulator (around 63 hours of base autonomy). Its runtime would be shortened for less than 1 minute, 16–66 minutes, or 16–45 minutes in respective scenarios on the account of application-specific processing and communication. This estimate excludes any power spent by a dedicated asthma diary mobile application on data visualization and user interaction, which may put additional heavy power burden on the smartphone's LCD screen.

6. Conclusion

With power consumption being one of critical parameter in designing wearable sensor systems, the goal of this analysis was to provide generalized knowledge on how different analog/digital processing hardware architectures and communication technologies handle the specific task of wearable asthmatic wheeze detection. The analysis covered three typical operating scenarios differing with respect to distribution of signal digitization, processing, and communication across the wearable sensor system. In addition to exact power figures reflecting the current state of the technology, analysis provided relative relations between different hardware architectures and component families, which can be used to guide similar and future designs in the high-impact field of wearable physiological signal monitoring devices.

Given the processing-intensive application, it is shown that the minimal end-to-end system power consumption (both sensor and smartphone) is achieved by implementing processing onboard the wearable sensor (328 to 428 μW). This scenario minimizes the power spent on communication, and the bottleneck consumer is the sensor's signal processing subsystem. The comparison of the proposed respiratory sound classification algorithms (HMM-based algorithm from [23] and referent frequency-tracking algorithm [12]) has shown that both algorithms feature comparable execution times. However, HMM-based algorithm lowers signal digitization power requirements by operating with signal digitized at only 2 kHz and in lower SNR conditions [23].

The analysis of acoustic sensors, analog signal conditioning, and ADC architectures has confirmed that analog MEMS microphones feature the greatest power efficiency for our application, totaling about 100 μW including the proposed analog signal conditioning circuit. For signal digitization, 16-bit successive approximation (SAR) ADC architecture proved optimal. Power analysis of the processing subsystem has shown that 32-bit ARM Cortex M3/M4 cores embedded within the Bluetooth 4 SoC modules feature the optimal trade-off between performance and power consumption.

Application of CS enables for construction of the system featuring a wearable sensor consuming minimal power (216 to 357 μW). This poses great implications on manufacturing costs of the wearable sensor as it enables for simpler (thus cheaper) hardware/firmware design. Also, it lowers the software development and maintenance and updates costs, by moving most of the software development from the sensor to the smartphone (i.e., from the domain of specialized MCU-/DSP-embedded firmware development, to common Android Java mobile application development). The biggest merit of the CS in comparison to the conventional compression techniques is that it minimizes the sensor's power, while simultaneously keeping the total system power lower or equal to the uncompressed streaming scenario. CS asymmetrically redistributes power load, redirecting it from sensor's acquisition, processing, and communication subsystems into the smartphone's processing subsystem. On the smartphone, CS reconstruction is the major bottleneck, costing in the case of OMP algorithm 2.2 to 5.4 times more than the classification. A single low-power ARM Cortex-A53 processing core running at the minimum frequency (e.g., 400 MHz) suffices for real-time CS reconstruction and classification.

In the next step, collected knowledge shall be used for advancing to the next step of technological readiness, entailing a construction of an integrated hardware prototype suitable for trials on patients. Specific engineering challenges expected on this path may include some of general limitations related to acoustic sensing modality: sensitivity to mechanical coupling [9, 40], external interferences (e.g., speech, heart sounds [4]), and noise (sensor contact noise, background noise [5]).

Abbreviations

ADC: Analog-to-digital converter
CS: Compressed sensing/compressive sampling
DCT: Discrete cosine transform
DFT: Discrete Fourier transform
DMA: Direct memory access
DSP: Digital signal processor
HMM: Hidden Markov model
I^2C: Interintegrated circuit bus
MCU: Microcontroller unit
OMP: Orthogonal matching pursuit
STFT: Short-term Fourier transform
PLL: Phase-locked loop
RX: Receive
SAR: Successive approximation
SoC: System-on-chip
SPI: Serial peripheral interface bus
TX: Transmit.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Both Vedran Bilas and Dinko Oletic conceived the research and designed the experiments; Dinko Oletic performed the experiments and analyzed the data; Dinko Oletic and Vedran Bilas wrote the paper. Dinko Oletic and Vedran Bilas contributed equally to this work.

References

[1] M. Masoli, D. Fabian, S. Holt, and R. Beasley, *Global Burden of Asthma*, Medical Research Institute of New Zealand, Wellington, New Zealand, 2010.

[2] H. Melbye, L. Garcia-Marcos, P. Brand, M. Everard, K. Priftis, and H. Pasterkamp, "Wheezes, crackles and rhonchi: simplifying description of lung sounds increases the agreement on their classification: a study of 12 physicians' classification of lung sounds from video recordings," *BMJ Open Respiratory Research*, vol. 3, no. 1, article e000136, 2016.

[3] H. Pasterkamp, P. L. Brand, M. Everard, L. Garcia-Marcos, H. Melbye, and K. N. Priftis, "Towards the standardisation of lung sound nomenclature," *European Respiratory Journal*, vol. 47, no. 3, pp. 724–732, 2016.

[4] Z. Moussavi, "Fundamentals of respiratory sounds and analysis," *Synthesis Lectures on Biomedical Engineering*, vol. 1, no. 1, pp. 1–68, 2006.

[5] J. A. Fiz, J. Gnitecki, S. S. Kraman, G. R. Wodicka, and H. Pasterkamp, "Effect of body position on lung sounds in healthy young men," *Chest Journal*, vol. 133, no. 3, pp. 729–736, 2008.

[6] U. Koehler, V. Gross, C. Reinke, T. Penzel, and C. F. Vogelmeier, "Akustische Analyse nächtlicher bronchialer Obstruktionen," *Pneumologie*, vol. 56, no. 1, pp. 19–24, 2002.

[7] A. L. Boner, G. L. Piacentini, D. G. Peroni et al., "Children with nocturnal asthma wheeze intermittently during sleep," *Journal of Asthma*, vol. 47, no. 3, pp. 290–294, 2010.

[8] L. Bentur, R. Beck, M. Shinawi, T. Naveh, and N. Gavriely, "Wheeze monitoring in children for assessment of nocturnal asthma and response to therapy," *European Respiratory Journal*, vol. 21, no. 4, pp. 621–626, 2003.

[9] H. Pasterkamp, S. S. Kraman, and G. R. Wodicka, "Respiratory sounds. Advances beyond the stethoscope," *American Journal of Respiratory and Critical Care Medicine*, vol. 156, no. 3, pp. 974–987, 1997.

[10] S. A. Taplidou and L. J. Hadjileontiadis, "Wheeze detection based on time-frequency analysis of breath sounds," *Computers in Biology and Medicine*, vol. 37, no. 8, pp. 1073–1083, 2007.

[11] B. S. Lin, B. S. Lin, and H. D. Wu, "Wheeze recognition based on 2D bilateral filtering of spectrogram," *Biomedical Engineering: Applications, Basis and Communications*, vol. 18, no. 3, pp. 128–137, 2006.

[12] D. Oletic, B. Arsenali, and V. Bilas, "Low-power wearable respiratory sound sensing," *Sensors*, vol. 14, no. 12, pp. 6535–6566, 2014.

[13] B. S. Lin and T. S. Yen, "An FPGA-based rapid wheezing detection system," *International Journal of Environmental Research and Public Health*, vol. 11, no. 12, pp. 1573–1593, 2014.

[14] M. Wiśniewski and T. P. Zieliński, "Joint application of audio spectral envelope and tonality index in an e-asthma monitoring system," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, pp. 1009–1018, 2015.

[15] I. Mazić, M. Bonković, and B. Džaja, "Two-level coarse-to-fine classification algorithm for asthma wheezing recognition in children's respiratory sounds," *Biomedical Signal Processing and Control*, vol. 21, pp. 105–118, 2015.

[16] O. Boujelben and M. Bahoura, "FPGA implementation of an automatic wheezes detector based on MFCC and SVM," in *2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pp. 647–650, Monastir, Tunisia, March 2016.

[17] G. C. Chang, "FPGA realization of an automatic wheeze detector based on wavelet filter bank," in *2009 IEEE 13th International Symposium on Consumer Electronics*, pp. 832-833, Kyoto, Japan, May 2009.

[18] M. Bahoura, "Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes," *Computers in Biology and Medicine*, vol. 39, no. 9, pp. 824–843, 2009.

[19] M. Lozano, J. A. Fiz, and R. Jané, "Automatic differentiation of normal and continuous adventitious respiratory sounds using ensemble empirical mode decomposition and instantaneous frequency," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 2, pp. 486–497, 2016.

[20] L. J. Hadjileontiadis, *Lung Sounds: An Advanced Signal Processing Perspective*, Number 9 in Synthesis Lectures on Biomedical Engineering, Morgan & Claypool, 2008.

[21] D. Oletic, *Low-Power Wearable System for Asthmatic Wheeze Detection, [Ph.D. thesis]*, 2016.

[22] D. Oletic, M. Skrapec, and V. Bilas, "Hidden Markov model in spectro-temporal tracking of asthmatic wheezing in respiratory sounds," *6th European Conference of the International Federation for Medical and Biological Engineering. IFMBE Proceedings*, I. Lacković and D. Vasic, Eds., , pp. 5–8, Springer, Cham, 2015.

[23] D. Oletic and V. Bilas, "Asthmatic wheeze detection from compressively sensed respiratory sound spectra," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, p. 1, 2017.

[24] M. Wisniewski and T. Zielinski, "Digital analysis methods of wheezes in asthma," in *ICSES 2010 International Conference on Signals and Electronic Circuits*, pp. 69–72, Gliwice, Poland, September 2010.

[25] D. Oletic, B. Arsenali, and V. Bilas, "Towards continuous wheeze detection body sensor node as a core of asthma monitoring system," in *Wireless Mobile Communication and Healthcare. MobiHealth 2011*, K. S. Nikita, J. C. Lin, D. I. Fotiadis, and M. T. Arredondo Waldmeyer, Eds., vol. 83 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pp. 165–172, Springer, Berlin, Heidelberg, 2012.

[26] C. Yu, T. H. Tsai, S. I. Huang, and C. W. Lin, "Soft stethoscope for detecting asthma wheeze in young children," *Sensors*, vol. 13, no. 12, pp. 7399–7413, 2013.

[27] http://www.respiri.co/.

[28] K. Li, *Implementation of the CUSUM Algorithm on FPGA for Transient Signal Detection, [Ph.D. thesis]*, Auckland University of Technology, 2012.

[29] R. Sarpeshkar, "Universal principles for ultra low power and energy efficient design," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 4, pp. 193–198, 2012.

[30] D. Oletic and V. Bilas, "Energy-efficient respiratory sounds sensing for personal mobile asthma monitoring," *IEEE Sensors Journal*, vol. 16, no. 23, pp. 8295–8303, 2016.

[31] D. Oletic, M. Skrapec, and V. Bilas, "Prototype of respiratory sounds monitoring system based on compressive sampling," in *The International Conference on Health Informatics*, Y. T. Zhang, Ed., vol. 42 of IFMBE Proceedings, pp. 92–95, Springer, Cham, 2014.

[32] D. Oletic, M. Skrapec, and V. Bilas, "Monitoring respiratory sounds: compressed sensing reconstruction via OMP on android smartphone," in *Wireless Mobile Communication and Healthcare. MobiHealth 2012*, B. Godara and K. S. Nikita, Eds., vol. 61 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, pp. 114–121, Springer, Berlin, Heidelberg, 2013.

[33] T. Sakai, H. Satomoto, S. Kiyasu, and S. Miyahara, "Sparse representation-based extraction of pulmonary sound components from low-quality auscultation signals," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 509–512, Kyoto, Japan, March 2012.

[34] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[35] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*, Cambridge University Press, 2012.

[36] http://www.csr.com/products/aptx.

[37] R. Salami, C. Laflamme, B. Bessette, and J. P. Adoul, "ITU-T G. 729 Annex A: reduced complexity 8 kb/s CS-ACELP codec for digital simultaneous voice and data," *IEEE Communications Magazine*, vol. 35, no. 9, pp. 56–63, 1997.

[38] D. Oletic and V. Bilas, "Wireless sensor node for respiratory sounds monitoring," in *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, pp. 28–32, Graz, Austria, May 2012.

[39] A. R. A. Sovijarvi, J. Vanderschoot, and J. E. Earis, "Standardization of computerized respiratory sound analysis," *European Respiratory Review*, vol. 10, no. 77, p. 585, 2000.

[40] S. S. Kraman, G. R. Wodicka, G. A. Pressler, and H. Pasterkamp, "Comparison of lung sound transducers using a bioacoustic transducer testing system," *Journal of Applied Physiology*, vol. 101, no. 2, pp. 469–476, 2006.

[41] M. ZaÑartu, J. C. Ho, S. S. Kraman, H. Pasterkamp, J. E. Huber, and G. R. Wodicka, "Air-borne and tissue-borne sensitivities of bioacoustic sensors used on the skin surface," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 2, pp. 443–451, 2009.

[42] August 2016, http://www.analog.com/media/en/technical-documentation/obsolete-data-sheets/ADMP441.pdf.

[43] August 2016, http://www.farnell.com/datasheets/1780663.pdf?_ga=1.147712756.2140163270.1467027387.

[44] August 2016, http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL337.pdf.

[45] August 2016, http://www.farnell.com/datasheets/1794369.pdf.

[46] August 2016, http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf.

[47] August 2016, https://www.invensense.com/wp-content/uploads/2015/02/ICS-40310-datasheet-v1.2.pdf.

[48] August 2016, http://www.ti.com/lit/ds/sbos445c/sbos445c.pdf.

[49] August 2016, http://www.ti.com/lit/ds/symlink/ads7924.pdf.

[50] August 2016, http://www.analog.com/media/en/technical-documentation/data-sheets/AD7684.pdf.

[51] August 2016, http://www.ti.com/lit/ds/symlink/ads1015.pdf.

[52] August 2016, http://www.ti.com/lit/ds/symlink/ads1115.pdf.

[53] August 2016, http://www.ti.com/lit/ds/symlink/ads1251.pdf.

[54] P. Vidas, B. Jeren, P. Crnkovic, and I. Jovanovic, "Digital signal processing systems with low power consumption," in *[1991 Proceedings] 6th Mediterranean Electrotechnical Conference*, pp. 343–346, LJubljana, Slovenia, May 1991.

[55] L. Benini and G.d. Micheli, "System-level power optimization: techniques and tools," *ACM Transactions on Design Automation of Electronic Systems*, vol. 5, no. 2, pp. 115–192, 2000.

[56] August 2016, http://www.ti.com/lit/ds/symlink/tms320c5535.pdf.

[57] August 2016, http://cache.nxp.com/files/32bit/doc/ref_manual/MCF51MM256RM.pdf?fpsp=1&WT_TYPE=Reference%20Manuals&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation&fileExt=.pdf.

[58] August 2016, http://www.nxp.com/files/dsp/doc/data_sheet/MC56F8006.pdf?fsrch=1.

[59] August 2016, http://www.analog.com/media/en/technical-documentation/data-sheets/ADSP-2188M.pdf.

[60] August 2016, http://www.st.com/content/ccc/resource/technical/document/datasheet/66/71/4b/23/94/c3/42/c8/CD00277537.pdf/files/CD00277537.pdf/jcr:content/translations/en.CD00277537.pdf.

[61] August 2016, http://www.analog.com/media/en/technical-documentation/data-sheets/ADuC7060_7061.pdf.

[62] August 2016, http://www.nxp.com/documents/data_sheet/LPC1102_1104.pdf.

[63] August 2016, http://www.ti.com/lit/ds/symlink/msp430fr5729.pdf.

[64] August 2016, http://www.ti.com/lit/ds/symlink/cc2541.pdf.

[65] August 2016, http://www.mouser.com/pdfdocs/PRODBRIEF nRF51422.pdf.

[66] August 2016, https://www.silabs.com/Support%20Documents/RegisteredDocs/BGM113_DataSheet.pdf.

[67] August 2016, http://www.ti.com/lit/ds/symlink/cc2640.pdf.

[68] August 2016, http://infocenter.nordicsemi.com/pdf/nRF52832_PS_v1.0.pdf.

[69] J. G. Teeter and E. R. Bleecker, "Relationship between airway obstruction and respiratory symptoms in adult asthmatics," *Chest*, vol. 113, no. 2, pp. 272–277, 1998.

[70] C. Lenclud, G. Cuttitta, D. Van Gansbeke et al., "Evaluation of nocturnal bronchoconstriction by all night tracheal sound monitoring," *Thorax*, vol. 51, no. 7, pp. 694–698, 1996.

[71] D. E. Bellasi and L. Benini, "Energy-efficiency analysis of analog and digital compressive sensing in wireless sensors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 11, pp. 2718–2729, 2015.

[72] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao, and V. C. Leung, "Body area networks: a survey," *Mobile Networks and Applications*, vol. 16, no. 2, pp. 171–193, 2011.

[73] T. Instruments, *Application Report SWRA478A: Measuring Bluetooth Smart Power*, Texas Instruments, 2016, Technical report.

[74] August 2016, http://www.ti.com/ble-power-calculator.

[75] A. FrumusanuAugust 2016, http://www.anandtech.com/show/9330/exynos-7420-deep-dive/5.

[76] A. Frumusanu and R. Smith, August 2016, http://www.anandtech.com/show/8718/the-samsung-galaxy-note-4-exynos-review/4.

[77] August 2016, https://static.docs.arm.com/ddi0500/f/DDI0500.pdf.

[78] August 2016, https://static.docs.arm.com/ddi0488/g/DDI0488G_cortex_a57_mpcore_trm.pdf.

[79] J. Tropp and A. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

The Scientific World Journal

International Journal of Rotating Machinery

Journal of Sensors

Advances in Multimedia

Journal of Engineering

Advances in Civil Engineering

Journal of Control Science and Engineering

Journal of Robotics

Journal of Electrical and Computer Engineering

Hindawi

Submit your manuscripts at
www.hindawi.com

Advances in OptoElectronics

VLSI Design

International Journal of Navigation and Observation

Modelling & Simulation in Engineering

International Journal of Aerospace Engineering

International Journal of Chemical Engineering

International Journal of Antennas and Propagation

Active and Passive Electronic Components

Shock and Vibration

Advances in Acoustics and Vibration