Bluetooth-Enabled Transparent RF Sensing

Rong Ding, Haiming Jin, Ningzhi Zhu, Zijie Chen, Yi Fu, Fengyuan Zhu,
Guiyun Fan, Xiaohua Tian, Linghe Kong
{dingrong,jinhaiming,ningzhizhu,chenzj1,jerry_fuyi,jsqdzhufengyuan,fgy726,xtian,linghe.kong}@sjtu.edu.cn
Shanghai Jiao Tong University, Shanghai, China

Abstract

This paper presents Serafin, the first full-stack, sub-mW, and versatile Bluetooth-enabled RF sensor that brings transparent RF sensing to any mobile and IoT device: it independently conducts the whole RF sensing process from RF signal reception to sensing result computation in a wide variety of sensing tasks with only negligible power consumption. At the core of Serafin are our two designs that address the challenge posed by the stringent sub-mW power constraint to jointly achieving versatility and full stackness. Specifically, (i) we utilize the ambient Bluetooth advertising signal as the signal for sensing, and extract the phase difference of the sensing signals received by each antenna pair from the amplitude of their sum signal, which avoids power-hungry hardware components and intensive computation, and (ii) we employ low-power MCU as the computation hardware, and suppress its power consumption by activating it adaptively only when necessary and customizing a light-weight neural network model that still ensures satisfactory inference accuracy. Our extensive experiments on 6 representative sensing tasks show that Serafin achieves **competitive sensing performance**, but consumes only around **500-900μW** power, which is **3-4 orders of magnitude lower** than those of existing full-stack and versatile counterparts.

CCS Concepts

• **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; • **Hardware** → **Sensor devices and platforms**; **Wireless devices**.

Haiming Jin and Guiyun Fan are the corresponding authors.

ACM MOBICOM '25, Hong Kong, China

ACM ISBN 979-8-4007-1129-9/2025/11
https://doi.org/10.1145/3680207.3723468

Keywords

RF sensor, sub-mW, RF computing.

ACM Reference Format:
Rong Ding, Haiming Jin, Ningzhi Zhu, Zijie Chen, Yi Fu, Fengyuan Zhu, Guiyun Fan, Xiaohua Tian, and Linghe Kong. 2025. Bluetooth-Enabled Transparent RF Sensing. In *The 31st Annual International Conference on Mobile Computing and Networking (ACM MOBICOM '25), November 4–8, 2025, Hong Kong, China.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3680207.3723468

1 Introduction

Motivation. The past decade has witnessed the feasibility of empowering mobile and IoT devices with RF sensing capability via integrating them with dedicated RF sensors (e.g., mmWave radar [1]) or reusing their on-board communication modules (e.g., WiFi chipset [2]) as RF sensors. However, existing RF sensors typically incur excessive power consumption to their host devices. On the one hand, they are power-intensive (100mW- to W-level) themselves, and Google reportedly removed the Soli mmWave radar in its smartphone products later than Pixel 4 for such reason [3, 4]. On the other hand, they rely on their host devices to compute the ultimate sensing results (e.g., recognized gesture) based on their output data (e.g., I/Q samples), and the computation process alone could consume W-level power on common host devices (e.g., smartphone, laptop, mobile robot).

Clearly, it is desirable to design an RF sensor that is **transparent to mobile and IoT device**, i.e., one that brings RF sensing to host device but consumes negligible power, or equivalently, one that has the following characteristics.

- *Full-stack:* containing a whole suite of hardware and software that carries out the entire process of RF sensing from RF signal reception to sensing result computation.
- *Sub-mW:* consuming only sub-mW power, which is negligible to common host devices, and could alternatively be provided by low-capacity power sources, such as coin batteries or even harvested ambient energy.
- *Versatile:* supporting a wide variety of RF sensing tasks (e.g., gesture recognition, respiration sensing, localization) to meet the diverse sensing demands of host devices.

Challenge. Though promising, such RF sensor that jointly enjoys the aforementioned three characteristics does not exist yet: existing full-stack RF sensors either lack versatility as

they are tailored to only one or two specific types of sensing tasks [5, 6], or consume W-level power [7]. In fact, the sub-mW power consumption is a fundamental constraint that poses severe challenge to simultaneously achieving versatility and full stackness, and we explain the reason as follows.

To achieve versatility, an RF sensor should be able to extract the phase difference of the RF signals received by any antenna pair, as it is necessary for deriving the AoA and TDoA essential for common sensing tasks. However, there currently lack sub-mW phase difference extraction methods: existing methods employ superheterodyne, low-IF, or zero-IF receiver, whose power-hungry hardware components (e.g., VCO, LNA) and intensive computation processes (e.g., massive FFT) totally consume 10mW- to 100mW-level power.

Even if one manage to extract the phase difference with far below 1mW power consumption, it is still challenging to go one step further to achieve full stackness without exceeding the remaining power budget. Specifically, existing methods oftentimes compute sensing results with neural network models which require a minimum of 10MHz-level system clock frequency for timely inference. However, even a low-power micro-controller unit (MCU) operating at such system clock frequency could still consume mW-level power.

Philosophies behind Serafin. We design and implement Serafin[1] as shown in Fig. 1, the first full-stack, sub-mW, and versatile Bluetooth-enabled RF sensor that brings transparent RF sensing to mobile and IoT device. The design philosophies of Serafin are twofold and are elaborated as follows.

First, we realize phase difference extraction from a whole new perspective by drawing inspiration from the phenomenon that the amplitude of the sum signal of two co-frequency single-tone signals has a strict mapping to their phase difference. Specifically, Serafin utilizes the ambient near single-tone Bluetooth advertising signal as the signal for sensing, and *extracts the phase difference of the signals received by a pair of antennas from the amplitude of their sum signal* without relying on power-hungry analog components and intensive computation. In multipath-free environments, Serafin extracts the phase difference of signals propagating through the single line-of-sight path. Such phase difference is necessary for many sensing tasks requiring accurate AoA or TDoA, such as localization. In multipath-rich environments, the signal received by Serafin's each antenna is the superposition of the signals propagating through multiple paths, and the phase difference extracted by Serafin in such environments is that of the superposition signals. Although there lacks a direct mapping from such phase difference to the AoA and TDoA of the signals in line-of-sight path, the variation of such phase difference is closely related to the motion pattern of the sensed object, and is sufficient for sensing tasks

[1]Serafin comes from Sub-mW, versatile, and full-stack Rf sensing.

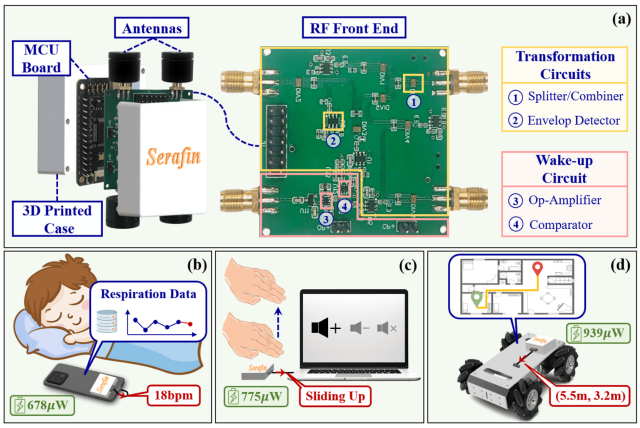


Figure 1: Serafin prototype (a), where only one among the same hardware components of its RF front end is marked with circled number. Serafin brings transparent RF sensing to mobile and IoT devices, such as smartphone (b), laptop (c), and mobile robot (d).

that require motion pattern recognition, such as human gesture and activity classification. Second, we select low-power MCU as the computation hardware, and suppress its power consumption by *accurately identifying every possible chance to deactivate the MCU*, and *customizing a light-weight neural network model* that ensures satisfactory inference accuracy.

Key techniques of Serafin. Serafin achieves the above design philosophies via the following three key techniques, including amplitude-based phase difference extraction, adaptive wake-up, and light-weight inference.

Amplitude-based phase difference extraction. This technique transforms the phase difference to amplitude via our specially designed analog transformation circuit, and computes the phase difference based on amplitude samples via only a few float point calculations. Specifically, the circuit takes in the signals received by two antennas, employs passive Wilkinson power splitter and combiner to construct the sum signal, and uses low-power envelope detectors to obtain the amplitude of the sum signal for phase difference computation. The circuit consumes only 100μW-level power.

Adaptive wake-up. This technique enables the ADC and MCU to wake up only when Bluetooth advertising signals appear via hardware-software co-design. From the hardware perspective, we design a wake-up circuit consisting of a low-power operational amplifier and noise-resistant comparator, which robustly detects the presence of RF signals. From the software perspective, we design a wake-up algorithm which identifies the Bluetooth advertising signals according to their unique packet and inter-packet duration. Due to the less than 2% duty cycle of Bluetooth advertising signals, this technique saves over 98% sampling and computation power compared with any design that keeps the ADC and MCU always on.

Light-weight inference. This technique suppresses the power consumption of the MCU on neural network inference.

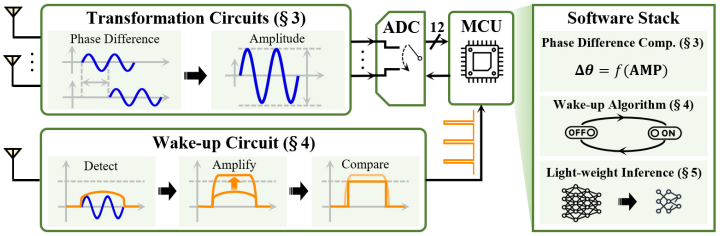


Figure 2: Serafin overview: hardware and software components.

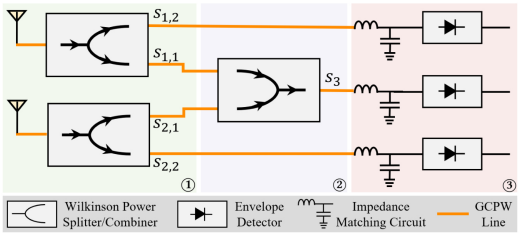


Figure 3: Transformation circuit.

Architecturally, we design a light-weight yet versatile neural network model, which primarily consists of a few simple 1D CNN layers and is powerful enough to handle the time-series inputs in common sensing tasks. Besides, we carefully quantize the model parameters to further alleviate the workload of the MCU but still ensure satisfactory inference accuracy. This technique reduces the average power consumption on sensing result computation to only 10μW-level.

Summary of contributions. In what follows, we summarize the main contributions of this paper in terms of functionality, technique, and implementation.

- *Functionality.* We propose Serafin, the first full-stack, sub-mW, and versatile Bluetooth-enabled RF sensor that brings transparent RF sensing to mobile and IoT device: Serafin stands out from existing sub-mW counterparts by simultaneously enjoying the merits of versatility and full stackness, and consumes 3-4 orders of magnitude lower power than those counterparts possessing these two merits.
- *Technique.* We address the core technical challenges posed by the sub-mW power constraint to versatility and full stackness via three specially designed techniques: (i) an amplitude-based phase difference extraction technique that helps achieve 100μW-level phase difference extraction, (ii) an adaptive wake-up technique that activates the ADC and MCU only when necessary, and (iii) a light-weight inference technique that realizes sensing result computation with only 10μW-level power consumption.
- *Implementation.* We prototype Serafin on a four-layer PCB and validate it on 6 representative sensing tasks. Serafin achieves (i) 94% accuracy for gesture recognition, (ii) 96% accuracy for activity recognition, (iii) 4% error for respiration sensing, (iv) 0.25m error for localization, (v) 8.3% error for refractive index measurement, and (vi) 97% accuracy for liquid classification. Serafin completes each task with only around 500-900μW power consumption.

2 Serafin Overview

Fig. 2 shows the hardware and software components of Serafin. We give an overview of Serafin in terms of its sensing signal, hardware, software, and workflow as follows.

Sensing signal. Serafin utilizes Bluetooth advertising signal at channel 37 for sensing, because it is approximately single-tone in the context of RF sensing. The signal utilizes Gaussian frequency shift keying modulation, where a binary bit is represented by a single-tone signal whose frequency is either positively or negatively shifted approximately 250kHz from the 2.402GHz center frequency of channel 37. Such 500kHz frequency shift is negligible for RF sensing, as it incurs only around 0.02% change in the signal frequency. Note that in each advertising event, the Bluetooth advertising packet is transmitted consecutively at channel 37, 38, and 39. Serafin can be designed to utilize the signal advertised at any single channel for sensing, whereas it cannot utilize the signals at more than one channel simultaneously because their frequencies change around 2% and cannot be ignored.

Hardware. Serafin's hardware consists of multiple transformation circuits, a wake-up circuit, an ADC, and an MCU. Each transformation circuit is connected to three sampling channels of the ADC. The wake-up circuit is connected to the GPIO port of the MCU.

Software. Serafin's software consists of the phase difference computation process, the wake-up algorithm, and the light-weight inference process executed by the MCU.

Workflow. Upon receiving RF signal that reaches its sensitivity, the wake-up circuit generates a rising voltage edge to the MCU. If a pre-defined length of time has already elapsed since the previous reception of Bluetooth advertising signal, the MCU runs the wake-up algorithm to identify whether the newly received signal originates from Bluetooth advertising. A negative identification result will immediately send the MCU back to sleep; otherwise, the MCU will control the ADC to sample the amplitudes output by the transformation circuits. The sampled amplitudes are then utilized by the phase difference computation process to calculate the phase difference. Finally, Serafin yields the sensing results based on the phase difference via the light-weight inference process.

Next, we present the key techniques of Serafin incorporated in the workflow: the *amplitude-based phase difference extraction* technique (§ 3), the *adaptive wake-up* technique (§ 4), and the *light-weight inference* technique (§ 5).

3 Amplitude-based Phase Difference Extraction Technique

The intuition behind the amplitude-based phase difference extraction technique is to transform the phase difference of the Bluetooth advertising signals received by an antenna pair

to the amplitude of their sum signal without using power-hungry components, and obtain the phase difference based on the amplitude samples with simple computation. Specifically, such technique consists of a *transformation circuit* (§ 3.1) and a *phase difference computation process* (§ 3.2).

3.1 Transformation Circuit

Circuit design. The schematic of the transformation circuit is illustrated in Fig. 3. Two antennas are connected respectively to the input ports of two Wilkinson power splitters. Each splitter has one output port connected to one input port of a Wilkinson power combiner. The other output ports of the two splitters, and the output port of the combiner are connected respectively to an impedance matching circuit followed by an envelope detector. The key implementation choices of the circuit are described as follows.

Splitters&combiner. We implement each splitter and the combiner respectively using a compact and passive component PD2425J5050S2HF [8]. Specifically, such component is a three-port network used in the forward direction as a splitter and in the backward direction as a combiner.

Transmission lines. We implement the transmission lines connected to the splitters and combiner using the grounded coplanar waveguide (GCPW) lines, which are narrower than conventional microstrip lines with the same characteristic impedance, and thus desirable in terms of form factor.

Impedance matching. To enable the splitters and combiner to work normally, the characteristic impedance of the GCWP lines, and the overall impedance of each impedance matching circuit and its subsequent envelope detector should be tailored to 50Ω. We achieve this by adjusting the width of the GCPW lines using TXLINE [9], and deciding the values of the capacitor and inductor using ADS [10].

Envelope detectors. We choose LTC5530 [11] as the envelope detectors in this circuit for its satisfactory linearity.

In-circuit signal flow. We describe the signal flow in the transformation circuit, including the following three stages.

- *Stage 1 (split).* As shown in Fig. 3a, the signal $s_i(t)$ received by each antenna i is split into $s_{i,j}(t) = A_{i,j}\cos(f_c t+\theta_i+\phi_{i,j})$ with $i, j \in \{1, 2\}$, where f_c is the center frequency of the advertising channel, θ_i is the initial phase of $s_i(t)$, $A_{i,j} = \alpha_{i,j}A_i$ is the amplitude of $s_{i,j}(t)$, A_i is the amplitude of $s_i(t)$, and $\alpha_{i,j}$ and $\phi_{i,j}$ are the attenuation coefficient and phase delay caused by splitting. Note that in multipath-free environments, $s_i(t)$ is the received Bluetooth advertising signal propagating through a single path. In multipath-rich environments, $s_i(t)$ is a superposition of the Bluetooth advertising signals propagating through multiple paths.[2].
- *Stage 2 (combine).* As shown in Fig. 3b, the two signals $s_{1,1}(t)$ and $s_{2,1}(t)$ are combined into $s_3(t) = A_3\cos(f_c t + \theta_3)$, where θ_3 is the initial phase of $s_3(t)$, and A_3 is the amplitude of $s_3(t)$ given by the following Eq. (1),

$$A_3 = \sqrt{\sum_{i=1}^{2}(p_i A_{i,2})^2 + 2\Big(\prod_{i=1}^{2} p_i A_{i,2}\Big)\cos(\Delta\theta+\psi)}, \tag{1}$$

with $\Delta\theta = \theta_1 - \theta_2$, $\psi = \phi_{1,1} - \phi_{2,1} + \psi_1 - \psi_2$, $p_i = \alpha_{i,1}\beta_i/\alpha_{i,2}$, and ψ_i and β_i representing the phase delay and attenuation coefficient caused by the combining process.
- *Stage 3 (measure).* As shown in Fig. 3c, the amplitudes of $s_{1,2}(t)$, $s_{2,2}(t)$, and $s_3(t)$, namely $A_{1,2}$, $A_{2,2}$, and A_3, are output by the three envelope detectors and utilized to compute the phase difference $\Delta\theta$, as introduced in § 3.2.

3.2 Phase Difference Computation Process

Online execution. After Serafin is deployed to execute sensing tasks, the phase difference computation process running in its MCU computes $\Delta\theta$ based on the sampled amplitudes $A_{1,2}$, $A_{2,2}$, and A_3 according to Eq. (2),

$$\Delta\theta = \arccos\Big(\frac{A_3^2 - (p_1A_{1,2})^2 - (p_2A_{2,2})^2}{2p_1p_2A_{1,2}A_{2,2}}\Big) - \psi, \tag{2}$$

which is derived from Eq. (1). Note that since the co-domain of the $\arccos(\cdot)$ function used in Eq. (2) is $[0, \pi]$, the co-domain of the phase difference extracted by Serafin falls in $[-\psi, \pi-\psi]$ rather than $[-\pi, \pi]$. Thus, Serafin cannot obtain the correct phase difference if it falls in $[-\pi, -\psi) \cup (\pi - \psi, \pi]$. To avoid such issue, we carefully place the antennas of Serafin to be close enough and use delay transmission lines to make sure the actual phase difference falls in $[-\psi, \pi-\psi]$.

The prerequisite for such computation is to know p_1, p_2, and ψ. If the splitters and combiner are perfectly matched, the attenuation coefficients $\alpha_{i,1} = \alpha_{i,2} = \beta_i = \sqrt{2}/2$ and the phase delays $\phi_{1,1} = \phi_{2,1} = \psi_1 = \psi_2 = \pi/2$, and thus we have $p_1 = p_2 = \sqrt{2}/2$ and $\psi = 0$. In practice, the actual values of the capacitors and inductors in the impedance matching circuits have small yet non-negligible deviations from their nominal ones. Such deviations cause mismatches to the splitters and combiner, making p_1, p_2, and ψ unknown a priori but fixed after the transformation circuit is built. Thus, we propose to pre-measure these values offline before Serafin's deployment.

Offline pre-measuring. As shown in Fig. 4, before deploying Serafin, we conduct the following two offline pre-measuring steps to obtain the value of p_1, p_2, and ψ.

- *Step 1 (obtain p_1 & p_2).* To obtain p_1, we synthesize Bluetooth advertising signal using an HM8135 [12] RF synthesizer and inject the synthesized signal to antenna port 1 and leave antenna port 2 isolated with no input. Then, we collect the amplitudes measured by the middle and top envelope detectors in Fig. 3, and divide the former by the latter to obtain p_1, because they are respectively $\alpha_{1,1}\beta_1$ and $\alpha_{1,2}$ times the amplitude of the input signal in antenna port 1. We obtain p_2 with a similar procedure.

[2]In such case, Serafin extracts the phase difference of superposition signals.

- *Step 2 (obtain ψ).* We obtain ψ by injecting two synthesized signals with a priori known phase difference to antenna port 1 and port 2 respectively, and solves ψ according to Eq. (2) based on p_1 and p_2 obtained in step 1.

4 Adaptive Wake-up Technique

The intuition behind the adaptive wake-up technique is to detect the presence of RF signals via a low-power circuit and identify the Bluetooth advertising signals according to their unique packet duration and inter-packet duration. Specifically, such technique consists of a *wake-up circuit* (§ 4.1) and a *wake-up algorithm* (§ 4.2) described in details as follows.

4.1 Wake-up Circuit

Overall circuit design. As illustrated in Fig. 5, the wake-up circuit has a receiving antenna followed by an impedance matching circuit, and its three key components: a signal detector whose output voltage increases with the strength of the received RF signal, an amplification circuit which amplifies the output voltage of the signal detector, and a comparison circuit which generates a rising voltage edge to inform the MCU the receiving of RF signal.

Signal detector. As shown in Fig. 5a, the signal detector consists of a Schottky diode pair and a capacitor. Compared with the commercial envelop detector in the transformation circuit, such signal detector enjoys the merit of lower power consumption and is desirable for an always-on wake-up circuit. The implementation choices are described as follows.

Diodes. We implement the diode pair using the SMS7630-005LF [13] component, because it has a low forward voltage drop and fast switch speed, which facilitates the signal detector to detect the appearance of weak RF signals.

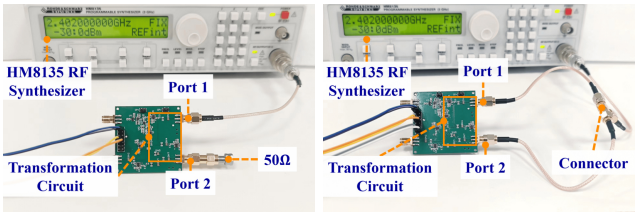
Capacitor. The value of the capacitor is set as $33p$F, because such value is sufficiently large to ensure that the signal detector could output a stable positive voltage when the wake-up circuit receives RF signal, and is meanwhile not excessively large to ensure that the output voltage of the signal detector quickly drops to zero when RF signal disappears.

Amplification circuit. As shown in Fig. 5b, the amplification circuit consists of an operational amplifier (op-amp) and two resistors R_1 and R_2. The implementation choices of the circuit are described as follows.

Amplification gain. We set the amplification gain of the circuit to be around 100. When Bluetooth advertising signal appears, such gain is sufficient to amplify the $1m$V-level output voltage of the signal detector, so as to trigger the comparison circuit by exceeding its $100m$V reference voltage.

Resistors. We set $R_1 = 10k\Omega$ and $R_2 = 1M\Omega$ to ensure the amplification gain $1 + R_2/R_1 \approx 100$.

Op-amp. We select MAX9620 [14] as the op-amp, because it has large gain-bandwidth product (GBWP) among those with similar power consumption. Given the amplification



(a) Step 1 (obtain p_1 and p_2). (b) Step 2 (obtain ψ).

Figure 4: Pre-measuring process of phase difference computation.

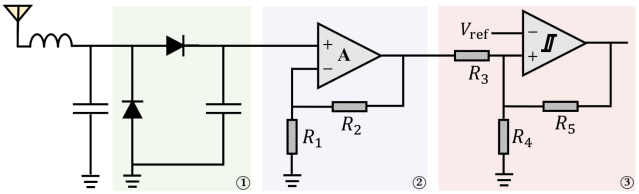


Figure 5: Wake-up circuit.

gain of 100, MAX9620 is able to amplify the input signal below $1.5M$Hz$/100 = 15k$Hz with almost no distortion. In fact, less distortion of the amplified signal indicates smaller delay for detecting Bluetooth advertising signal, faster wake-up of the MCU and ADC, as well as longer time reserved for ADC sampling before Bluetooth advertising signal disappears. In Fig. 6, we illustrate the benefits of larger GBWP by showing that the detection delay of the amplification circuit with MAX9620 as its op-amp is much lower than that with RS8521 [15] whose GBWP is only 1/4 of MAX9620.

Comparison circuit. As shown in Fig. 5c, the comparison circuit consists of a comparator and three resistors R_3, R_4, and R_5. A reference voltage $V_{\text{ref}} = 100m$V is provided to the negative input port of the comparator to eliminate the undesired trigger caused by its internal noise.

Comparator. We select MAX9075 [16] as the comparator for its ultra-low power consumption of only 9μW.

Resistors. The three resistors are introduced in the comparison circuit to eliminate the output jitter, which is caused by the noisy output voltage of the amplification circuit. Such voltage fluctuates around V_{ref}, making the output of the comparator frequently jitter, as illustrated in Fig. 7. The three resistors are set as $R_3 = 130k\Omega$, $R_4 = 750k\Omega$, and $R_5 = 10M\Omega$ to eliminate the output jitter by empowering the comparison circuit with $\Delta V = 20m$V hysteresis. As shown in Fig. 7, only if the input voltage of the comparison circuit (V_{in}) increases from $V_{\text{ref}} - \Delta V$ to $V_{\text{ref}} + \Delta V$, the output voltage of the comparison circuit (V_{out}) switches from low to high, and vice versa. Our choice of ΔV ensures a reliable jitter elimination since its value is one order larger than the noise voltage.

4.2 Wake-up Algorithm

Necessity. We design a wake-up algorithm that identifies whether the RF signal received by the wake-up circuit originates from Bluetooth advertising, and only wakes up the ADC to sample the amplitudes and the MCU to compute the

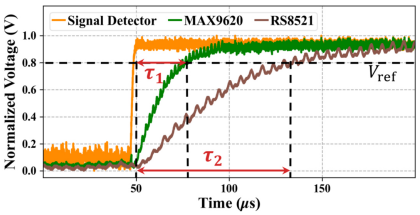


Figure 6: Detection delays of two op-amps.

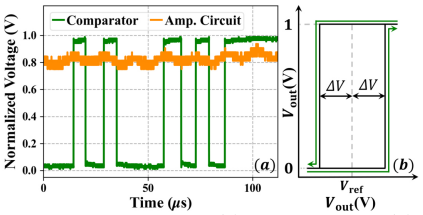


Figure 7: Output jitter (a) and hysteresis (b).

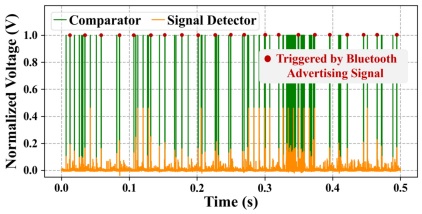


Figure 8: Impact of other signals.

phase difference when there is a positive identification result. Such algorithm is necessary in that other signals (e.g., WiFi) that commonly exist in real-world environments can also be detected by the wake-up circuit and frequently trigger it to generate rising voltage edges, which would unnecessarily wake up the ADC and MCU, resulting in power wastage. To illustrate such phenomenon, we deploy Serafin in a room with a Bluetooth beacon that generates Bluetooth advertising signals, as well as other communication devices such as WiFi AP and smartphones. We show the output of the signal detector and comparison circuit of the wake-up circuit in 0.5s in Fig. 8, which shows that signals other than Bluetooth advertising could frequently drive the comparison circuit to output rising voltage edges.

Intuition. The intuition behind the wake-up algorithm is to identify Bluetooth advertising signal based on its unique packet duration and inter-packet duration.

- *Packet duration.* A Bluetooth advertising packet has a payload of 31 octets, and thus a duration of around 400μs.
- *Inter-packet duration.* The inter-packet duration of two adjacent Bluetooth advertising packets at channel 37 is a random value that belongs to the set $\mathcal{T} = \{t_f + n \times 1ms : 0 \le n \le 10, n \in \mathbb{Z}\}$, where t_f is predefined and fixed.

Workflow. We describe the workflow of the wake-up algorithm using a finite-state machine with four states as illustrated in Fig. 9, where each state is represented by a tuple (S_1, S_2) with S_1 and S_2 representing the state of the MCU and ADC, respectively. We describe the four states and the state transitions happening at each state as follows.

(*REC, SLP*). When Serafin is powered up in an environment with ambient Bluetooth advertising signal, it enters this state, where the MCU is active for recording (REC), and the ADC is sleeping (SLP). If the wake-up circuit detects an RF signal before the elapse of a predefined duration $T = 10(t_f + 10ms)$, the algorithm assigns a unique index i to the detected signal, records its appearance time T_a^i, disappearance time T_d^i, as well as duration D_i, and remains in this state. When T elapses, the algorithm finds the set $\mathcal{D} = \{i : D_i \approx 400\mu s\}$ consisting of every index whose corresponding signal has a (near) 400μs duration. Next, the algorithm finds the set $\mathcal{S} = \{j : T_a^j - T_d^i \in \mathcal{T}, i \in \mathcal{D}, j \in \mathcal{D}, j > i\}$, and sets the disappearance time of the latest Bluetooth advertising signal as $T_d = \max_{j \in \mathcal{S}} T_d^j$. The algorithm then acquires the current time t, and sets a countdown time $\Delta T = \max(0, T_d + t_f - t)$ for the MCU's low-power timer (LPTIM). Serafin then enters (IACT, SLP).

(*IACT, SLP*). In this state, the MCU is inactive (IACT) and the ADC is sleeping. When the LPTIM times out and the wake-up circuit detects an RF signal, Serafin enters (IC, SLP).

(*IC, SLP*). In this state, the MCU is active for interval check (IC) and the ADC is sleeping. The algorithm assigns an index k to the signal just detected in (IACT, SLP) and records its appearance time T_a^k. If the interval $T_a^k - T_d \notin \mathcal{T}$, Serafin re-enters (IACT, SLP); otherwise, it enters (DC, ACT).

(*DC, ACT*). In this state, the MCU is active for duration check (DC) and the ADC is active (ACT) for sampling the amplitudes output by the transformation circuit. The algorithm records the disappearance time T_d^k of signal indexed by k. If the duration $T_d^k - T_a^k \approx 400\mu s$, the algorithm resets $T_d = T_d^k$, acquires the current time t, and resets the countdown time $\Delta T = \max(0, T_d + t_f - t)$ for the LPTIM. Serafin then enters (IACT, SLP) to wait for the next Bluetooth advertising signal.

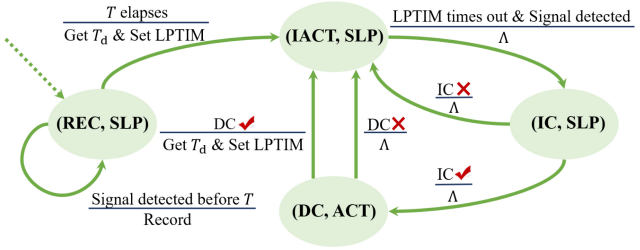


Figure 9: The finite-state machine of the wake-up algorithm. The initial state is pointed by the dashed arrow. $\frac{A}{B}$ indicates the algorithm executes B after event A happens. Λ indicates no operation.

5 Light-weight Inference Technique

The intuition behind the light-weight inference technique is to design a neural network model which is light-weight yet versatile enough for common RF sensing tasks, and carefully quantize the model parameters to suppress the inference workload of the MCU while maintaining inference accuracy. Specifically, such technique consists of *model design* (§ 5.1) and *parameter quantization* (§ 5.2) described as follows.

5.1 Model Design

Overview. We design SerafinNet, a light-weight neural network architecture which could handle the time-series inputs that commonly exist in RF sensing tasks. As illustrated in Fig. 10, SerafinNet consists of several cascading feature extraction blocks and an inference block.

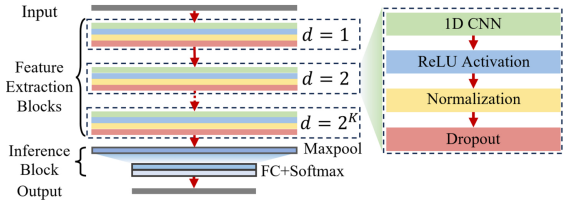


Figure 10: Model structure of SerafinNet with $K+1$ feature extraction blocks and d denoting the dilation rate of each block.

Feature extraction block. Such block consists of in series a 1D CNN layer, a ReLU activation layer, a normalization layer, and a dropout layer. The 1D CNN layer helps SerafinNet extract the temporal features. Compared with other model structures (e.g., LSTM or Transformer) for time-series input processing, 1D CNN has the advantage of fewer computation operations. For the k-th feature extraction block, we set the dilation rate of its 1D CNN layer as 2^{k-1}. Such design enables different blocks to extract temporal features at different scales. Besides, it also enables the extraction of global features with only a small number of feature extraction blocks, as the dilation rates of its 1D CNN layers rapidly grow exponentially. The ReLU activation layer introduces non-linearity into the network. The normalization layer stabilizes the training process by reducing the dependence of gradients on the scale of the inputs. The dropout layer prevents model overfitting in the training process.

Inference block. Such block contains a maxpooling layer, followed by a fully connected layer and a softmax layer. The output of the maxpooling layer maintains the dominant features of the input, while its size is only $1/m$ of that of the input, where m is the kernel size of the maxpooling layer. Such design reduces the amount of computation of the fully connected layer to its $1/m$.

5.2 Parameter Quantization

Overview. We apply the post-training quantization technique to SerafinNet to alleviate the computation workload of MCU while ensuring inference accuracy. The quantization process is implemented with Tensorflow Lite Converter [17].

Details. We utilize the dynamic quantization technique to quantize SerafinNet. This technique quantizes the weights of the 1D CNN layers and fully connected layers offline before inference, while calculates the best quantization coefficients for the activation layers and dynamically quantize them during inference. This technique enables to adapt the optimal quantization coefficients during inference. We quantize the 32-bit float parameters of SerafinNet with 8-bit integers, which is the highest quantization level that incurs negligible sacrifice of SerafinNet's inference accuracy.

6 Miscellaneous Design Details

Analog circuits. We integrate Serafin's analog circuits on a 5cm×5cm PCB. The analog circuits have 4 antennas, where one of them is used in the wake-up circuit, and the others are used in two transformation circuits. Among the latter three antennas, two of them have their received signals directly fed into one port of each transformation circuit, while the third one has its received signal split and fed into the other input ports of the two transformation circuits via a Wilkinson power splitter. The impact of the phase delays introduced in splitting is eliminated by our pre-measuring method in § 3.2.

MCU mode selection. Serafin's MCU is STM32L476RG [18], which has four inactive modes including sleep, stop, standby, and shutdown, and various active modes with different system clock frequencies. We choose the operation modes for the MCU's states introduced in § 4.2 as follows.

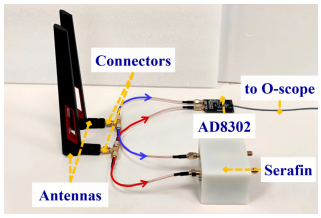
- When MCU is in IACT, we set it in sleep mode. Although the MCU consumes less power in the other inactive modes, exiting them requires re-initializing ADC and GPIO, which consumes ms-level time, making it problematic to enter state IC before Bluetooth advertising signal disappears. In contrast, exiting sleep mode does not require re-initializing ADC and GPIO, and consumes only μs-level time.
- When the MCU is in REC, IC, or DC, we set it in the active mode with a system clock frequency of $2M$Hz, which is the lowest frequency that enables the MCU to execute the wake-up algorithm, drive ADC sampling, and store the samples before Bluetooth advertising signal disappears.
- When Serafin requires the MCU for inference, we set it in the active mode with a system clock frequency of $80M$Hz to ensure $10m$s-level low inference latency.

7 Evaluation

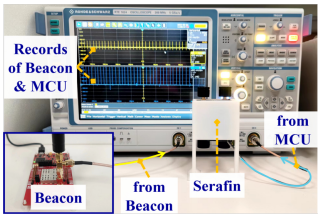
7.1 Overview and Signal Sources

We test Serafin from the perspectives of both micro-benchmarking and in-situ evaluation. In § 7.2, we benchmark the impact of received advertising signal strength on Serafin's phase difference extraction accuracy and wake-up performance. In § 7.3, we perform in-situ evaluation for Serafin on 6 representative RF sensing tasks, namely gesture recognition, activity recognition, respiration sensing, localization, refractive index measurement, and liquid classification.

We test Serafin on all the 6 sensing tasks (in § 7.3.1-§ 7.3.6) and micro-benchmarking evaluation with a TI CC1352P-2 [19] Bluetooth beacon as the signal source. Besides, we also test Serafin on the 3 human sensing tasks with a Thinkpad X1 Carbon [20] laptop or a Xiaomi 8 SE [21] smartphone as the signal source (in § 7.3.7). We aim to evaluate whether the laptop or smartphone could provide ambient Bluetooth advertising signals for Serafin in environments where no beacon exists, even though their transmission power is weaker than that of the beacon. We do not test Serafin with laptop or smartphone as the signal source for the other 3 sensing tasks, because only the beacon supports to connect a directional antenna required in refractive index measurement and



(a) Phase difference extraction.



(b) Wake-up evaluation.

Figure 11: Setups for micro-benchmarking.

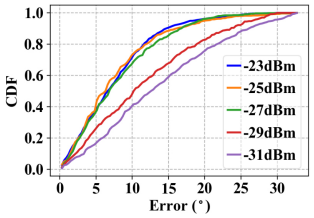


Figure 12: Phase difference error.

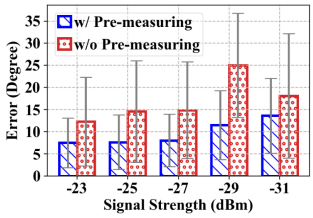


Figure 13: Pre-measuring impact.

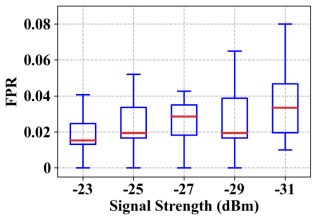


Figure 14: Wake-up FPR.

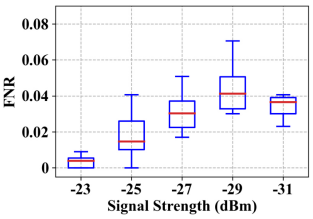


Figure 15: Wake-up FNR.

liquid classification, and the beacon is more suitable than the laptop and smartphone as the signal source for localization because its stronger advertising signal makes localization feasible in larger areas. All experiments have followed the standard procedures of IRB of our institute.

7.2 Micro-benchmarking

7.2.1 *Phase Difference Extraction Accuracy.*

Setup. The experimental setups for Serafin's phase difference extraction evaluation is shown in Fig. 11a.

Ground truth. We use 2 antennas to receive the Bluetooth advertising signals transmitted by the beacon. Each antenna is connected with a connector, which splits the received signal into two copies. The two splitted signals of each antenna are respectively injected into the transformation circuit and a dedicated phase difference detector AD8302 [22] with two equal-length cables. As such, the two signals injected into the transformation circuit have the same phase difference as those injected into AD8302. We use the measurements of AD8302 as the ground truth.

Signal strength. We vary the distance from the beacon to the antennas from 3.5m to 1.5m, and test Serafin's phase difference extraction accuracy under 5 receiving signal strength levels from -31dBm to -23dBm in a 2dBm increment.

Result. We present Serafin's overall phase difference extraction accuracy and the impact of pre-measuring.

Overall accuracy. As shown in Fig. 12, the median of Serafin's phase difference extraction error is persistently lower than 13° under different receiving signal strengths. As we will show in § 7.3, such phase difference extraction results can support the completion of many RF sensing tasks.

Impact of pre-measuring. As shown in Fig. 13, the errors of phase difference extraction with theoretical coefficients (i.e., $p_1 = p_2 = \sqrt{2}/2$ and $\psi = 0$ in Eq. (2)) are 1.4 to 2.2 times of those with pre-measuring. The reason is that pre-measuring obtains actual values of p_1, p_2, and ψ necessary for phase difference extraction in a practical transformation circuit.

7.2.2 *Wake-up Performance.*

Setup. The wake-up evaluation setup is shown in Fig. 11b.

Ground truth. The antenna port of the Bluetooth beacon is connected with a connector which connects with an antenna for transmitting advertising signals and an oscilloscope for recording the time of each transmission. We program Serafin's MCU to generate a rising voltage edge when Serafin enters (DC, ACT), in which ADC sampling and the phase difference computation process are executed. Such rising voltage edge is also collected by the oscilloscope. In this way, we obtain the ground-truth time when Bluetooth advertising signal appears and when Serafin enters (DC, ACT).

Other signals. To evaluate Serafin's wake-up performance with the existence of other signals, we place a Bluetooth keyboard, a Bluetooth mouse, and a smartphone near Serafin. The keyboard and mouse are connected to and interact with a laptop. The smartphone communicates with a WiFi AP at a data rate of around 800Kbps for downloading videos.

Performance. We measure the wake-up performance by two metrics: the false positive rate (FPR) and false negative rate (FNR). FPR is the ratio of Serafin's entering (DC, ACT) triggered by signals other than Bluetooth advertising ones. FNR is the ratio of Bluetooth advertising signals that fails to make Serafin enters (DC, ACT). We illustrate the FPR and FNR under different Bluetooth advertising signal strength levels in Fig. 14 and 15. The median of FPR and FNR in all experiments are persistently lower than 0.042. On the one hand, Serafin's wake-up algorithm can distinguish other signals from Bluetooth advertising signals, and thus contributes to low FPR. On the other hand, Serafin's wake-up circuit can detect weak Bluetooth advertising signals through amplification, and thus contributes to low FNR.

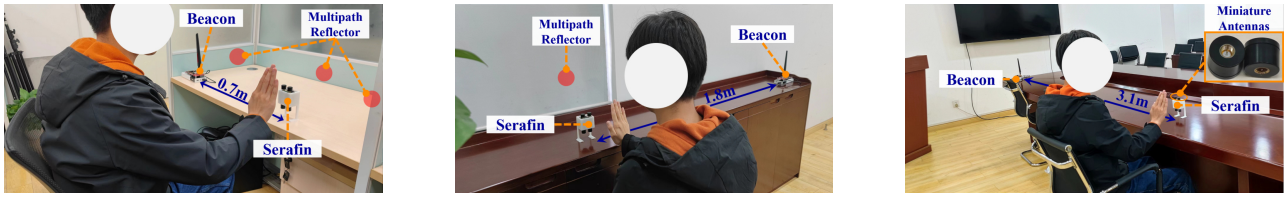
7.3 In-situ Evaluation

7.3.1 *Gesture Recognition.*

Setup. We present the setups of this task as follows.

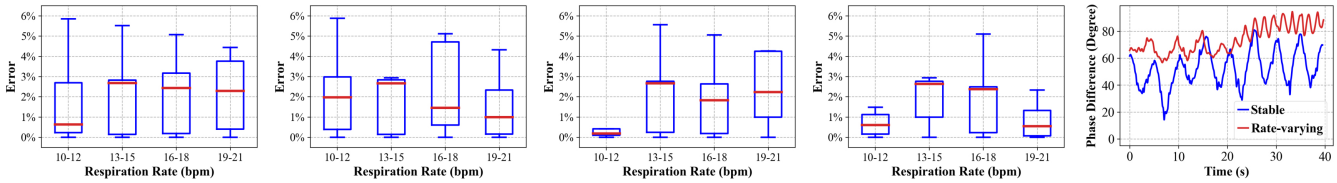
Gestures. This task aims to classify 8 representative human gestures, including sliding {left, right, up, down}, pushing {forward, back}, and shaking {left-right, up-down}.

Human. We employ 10 volunteers (denoted by H1-H10) for this task. These volunteers consist of both male and female, and have diverse heights, weights, and body shapes.



(a) Environment with rich multipath. (b) Environment with moderate multipath. (c) Environment with light multipath.

Figure 16: Experimental setups for gesture recognition: rich (a), moderate (b), and light (c) multipath interference.



(a) Overall error. (b) Error in E1. (c) Error in E2. (d) Error in E3. (e) Phase differences.

Figure 17: Respiration sensing results: overall error (a), errors in different environments (b-d), and phase difference sequences visualization (e).

Environment. As illustrated in Fig. 16, the experiments are conducted in 3 environments (denoted by E1-E3) with rich, moderate, and light multipath interference, respectively.

Dataset. Serafin is placed 0.5m away from the humans and collects the training and testing data. Each data instance is collected in 2s during which a gesture is performed, and contains the following time-series features, namely the phase difference sequences calculated by Serafin, as well as the amplitude sequences output by the two transformation circuits' top and bottom envelope detectors as shown in Fig. 3.

Result. We present the results as follows.

Overall accuracy. We randomly select 70% data instances to train SerafinNet, and use the remaining ones for testing. Serafin achieves an average accuracy of 94%.

Impact of phase difference. We remove the features of phase difference sequences in each data instance, and train, quantize, and test SerafinNet with such data instances. The gesture recognition accuracy decreases to 80%. Such results validate the necessity of phase difference on recognizing gestures.

Generalization. We evaluate the generalization capability of SerafinNet to recognize the gestures of unseen humans and environments. Specifically, we train 3 SerafinNets with data instances collected from human $\{H1, \cdots, H8\}$ in environments $\{E1, E2\}$, $\{E2, E3\}$, and $\{E1, E3\}$, respectively, with a representative generalization-enhancing training method [23]. Next, we quantize each model, and test it with all the data instances collected on humans or environments unseen in its training data. All the 3 models have over 90% accuracy.

7.3.2 *Activity Recognition.*

Setup. This task aims to classify 5 representative activities, including sitting, chair rotating, standing up, writing, and walking. The human subjects, environments, dataset construction procedures, training and testing methods are the same as those of the gesture recognition task, except that each data instance is collected in 3s.

Result. We present the results as follows.

Overall accuracy. The average accuracy of Serafin is 96%.

Impact of phase difference. When SerafinNet is trained using the data instances without the features of phase difference sequences, its accuracy on activity recognition decreases to 79%. Such results validate the necessity of phase difference on recognizing activities.

Generalization. We train, quantize, and test 3 SerfinNets following the same strategies as described in generalization test of the gesture recognition task. All of the 3 models have over 92% activity recognition accuracy.

7.3.3 *Respiration Sensing.*

Setup. This task aims to sense a human's respiration rate. We present the setups of this task as follows.

Human & environment. The humans and environments are the same as those in the gesture recognition task.

Dataset. Serafin is placed 0.5m from humans to collect data. Each data instance is collected in a 40s duration, containing the two phase difference sequences output by Serafin.

Ground truth. The ground truth of the respiration rate is obtained by a KIUZOU JZ-130R [24] fingertip pulse oximeter.

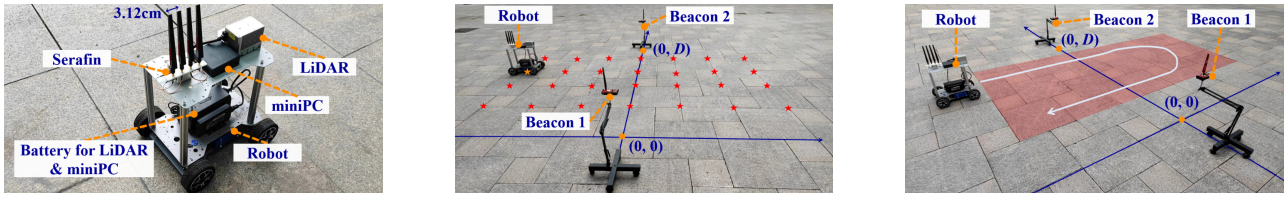
Method. We propose a simple yet effective method to calculate the respiration rate. The method smooths the two phase difference sequence, finds the numbers of local maximum in the two sequences, and outputs the average of the two numbers divided by 40s as the respiration rate.

Result. We present the results as follows.

Overall accuracy. As shown in Fig. 17a, the median errors of Serafin on respiration sensing are lower than 3% under different respiration ranges.

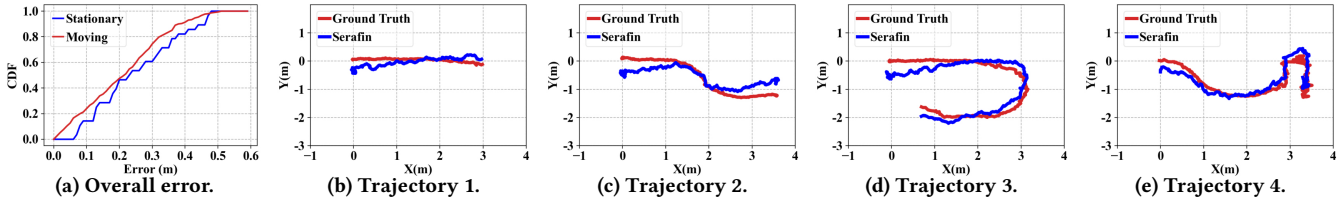
Performance in different environments. As shown in Fig. 17b-17d, the median error of Serafin are persistently lower than 3% in all the environments.

Phase difference sequence visualization. As shown in Fig. 17e, the extracted phase difference sequence shows a stable frequency when the respiration rate is stable, and varies with increasing frequencies when the respiration rate increases.



(a) Robot platform equipped with Serafin. (b) Stationary robot localization. (c) Moving robot localization.

Figure 18: Experimental setups for localization: robot platform (a), and localizing a stationary (b) and moving robot (c).



(a) Overall error. (b) Trajectory 1. (c) Trajectory 2. (d) Trajectory 3. (e) Trajectory 4.

Figure 19: Localization results: overall error (a) and visualization of representative trajectories (b-e).

7.3.4 *Localization.*

Setup. We present the setups of this task as follows.

Robot. This task aims to localize a mobile robot. As shown in Fig. 18a, the robot is equipped with Serafin to localize itself, a Livox Horizon LiDAR [25], and an Intel NUC 11 miniPC [26] that runs SLAM [27] to obtain the ground truth.

Antennas of Serafin. We arrange the antennas of Serafin's transformation circuits in a row, and the spacing between two adjacent antennas is 3.12cm, which is 1/4 of the wavelength of the Bluetooth advertising signals.

Mobility pattern. As shown in Fig. 18b and 18c, we conduct experiments for localizing a stationary or moving robot, respectively. In the stationary experiment, we place the robot at 28 locations evenly distributed in the 4m×2m area. In the moving experiment, we control the robot to move along 10 trajectories with various shapes.

Method. Serafin localizes the robot by measuring the AoA of the signals transmitted by the 2 beacons. Specifically, suppose the coordinates of the 2 beacons is $(0, 0)$ and $(0, D)$, the angle between the line of antenna array and line $y = 0$ is ζ_0, and the AoA of the received signals transmitted by the 2 beacons are ζ_1 and ζ_2, the coordinates of Serafin are thus $(\frac{D}{\arctan(\zeta_2-\zeta_0)+\arctan(\zeta_0-\zeta_1)}, \frac{D\arctan(\zeta_0-\zeta_1)}{\arctan(\zeta_2-\zeta_0)+\arctan(\zeta_0-\zeta_1)})$. In the stationary experiment, we place the antenna array to be parallel to line $y = 0$ and thus $\zeta_0 = 0$. In the moving experiment, an on-robot gyroscope is used to obtain the value of ζ_0.

Result. As illustrated in Fig. 19a, in experiments where the robot is stationary and moving, the median localization error of Serafin is only 0.25m and 0.24m, respectively. We also show the 4 representative trajectories tracked by Serafin and obtained by LiDAR SLAM in Fig. 19b-19e. The figures show that Serafin can faithfully track the shape of the trajectories.

7.3.5 *Refractive Index Measurement.*

Setup. This task aims to measure the refractive index (RI) of 6 ethanol solutions whose concentrations vary from 0% to 100% in a 20% increment. The setup is shown in Fig 20a. The ground truth is obtained from existing literature [28]. Each tested liquid is respectively filled in a plastic container. We attach 2 antennas of a transformation circuit on opposite sides of the container. The beacon uses a directional antenna to enhance the strength of the signal received by Serafin.

Method. We record the phase differences $\Delta\theta_1$ and $\Delta\theta_2$ of the signals received by the antennas before and after the container is filled with liquid. The liquid RI is calculated by $\frac{(\Delta\theta_2-\Delta\theta_1)\lambda}{2\pi d} + 1$, where $d = 0.5$cm is the liquid thickness and $\lambda = 12.5$cm is the signal wavelength.

Result. As illustrated in Fig 21a. The RI measurement error is persistently lower than 8.3%. This is attributed to Serafin's reliable performance on phase difference extraction.

7.3.6 *Liquid Classification.*

Setup. This task aims to classify 5 drinks, including water, juice, coke, wine, and milk. The setup is shown in Fig 20b. We attach Serafin's one of the three antennas in transformation circuits on one side of the bottle, and the other two antennas perpendicularly on the other side of the bottle.

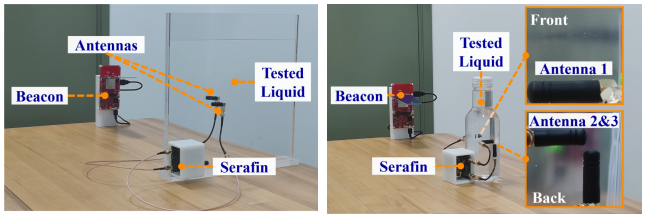
Method. We classify liquids based on two features, namely the phase difference between antenna 1 and 2, and the ratio of the amplitude of antenna 2 to that of antenna 3, as the former feature is influenced by the signal propagation speed in the liquid, and the latter feature reveals the influence the liquid has on signal polarization. We use k-means to obtain the classification result based on the above two features.

Result. As shown in Fig. 21b, the data samples for the same liquid type have similar values in both features. The liquid classification accuracy of Serafin is 97%.

7.3.7 *Laptop or Smartphone as the Signal Source.*

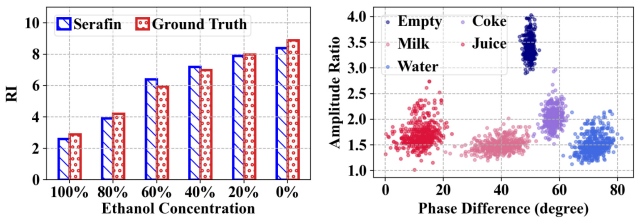
Setup. We present the setups as follows.

Serafin setup. The setup of Serafin with laptop or smartphone as the signal source is illustrated in Fig. 22. When laptop is the signal source, we use it as a beacon and place Serafin 1.5m away from it. When smartphone is the signal



(a) RI measurement. (b) Liquid classification.

Figure 20: Experimental setups for liquid sensing.



(a) RI measurement. (b) Liquid classification.

Figure 21: Experimental results for liquid sensing.

source, we integrate Serafin with the smartphone to enable the smartphone to sense nearby subjects with its own Bluetooth advertising signal. To minimize the form factor of Serafin-augmented smartphone, we customize four miniature and flexible patch antennas, attach them and Serafin on the back of the smartphone, and utilize a USB cable to convey the sensing results of Serafin to the smartphone.

Signal source setup. We aim to evaluate the impact of the signal source conducting communication tasks on Serafin's sensing performance, and the impact of sending advertising signal for Serfin on the communication performance of the signal source. To this end, we conduct experiments with the signal source in each of the five states: sending Bluetooth advertising signals (s1), transferring file to another laptop via Bluetooth (s2), connecting to a Bluetooth earbud for audio streaming (s3), running concurrent processes of Bluetooth file transfer and advertising (s4), and running concurrent processes of Bluetooth audio streaming and advertising (s5).

Result. We present the results as follows.

Accuracy w/o communication (s1). As illustrated in Tab. 1, Serafin achieves 5.8%, 4.7%, and 4.2% errors on gesture and activity recognition, and respiration sensing respectively with the signal source only sending advertising signals.

Impact of communication on sensing (s1, s4, & s5). Our experimental results indicate that for each of the three evaluated human sensing task, the sensing error increase brought by file transfer (s4 vs. s1) or audio streaming (s5 vs. s1) is lower than only 1.7%. Clearly, such results validate that the signal source conducting Bluetooth communication will not impair Serafin's sensing performance. The rationale behind such results lies in that Serafin can easily distinguish the advertising signals and communication signals via its wake-up algorithm, because these two kinds of signals are transmitted in a time division multiplexing (TDM) manner without

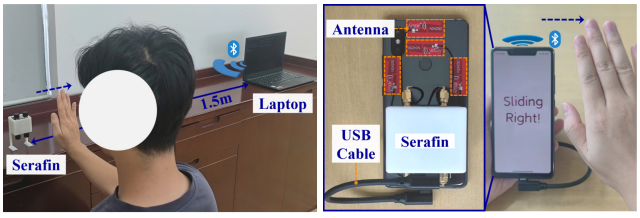


Figure 22: Laptop (left) and smartphone (right) as signal source.

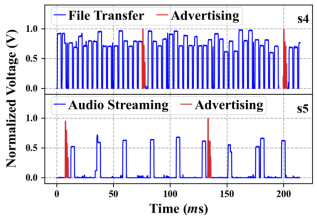


Figure 23: Bluetooth TDM.

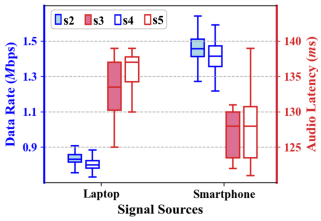


Figure 24: Comm. performance.

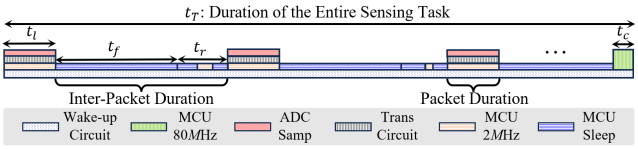


Figure 25: Power consumption breakdown.

overlap. We show a clear example of such TDM transmission in Fig. 23, which illustrates the output of the signal detector of Serafin when the signal source is in s4 or s5.

Impact of sensing on communication (s2-s5). As illustrated in Fig. 24, sending advertising signal for Serafin incurs only below 4% drop on file transfer data rate, and only below 2% increase on audio streaming latency. Such results validate that acting as the signal source of Serafin has negligible impact on normal communication of the laptop or smartphone.

7.4 Power Consumption Analysis

The average power consumption of Serafin's each component in a sensing task equals to its duty cycle throughout the entire sensing task multiplied by its operational power.

Duty cycle analysis. As illustrated in Fig. 25, when Serafin utilizes the Bluetooth advertising signals transmitted by one signal source for sensing, the duty cycle of its each individual component is as follows.

- The wake-up circuit keeps active and its duty cycle is 1.
- The system clock frequency of MCU reaches 80 *M*Hz at the end of the sensing task for computing the sensing results. The duty cycle of this mode is t_c/t_T, where t_c is the computation time, and t_T is the duration of the task.
- The ADC samples the outputs of the 6 envelope detectors in 2 transformation circuits only when Bluetooth advertising signal appears. Thus, its sampling rate[3] is approximately $\mathbb{E}[6/(t_f+t_r+t_l)]$, where $t_l = 400\mu s$ is the packet duration of Bluetooth advertising signal, t_f is the fixed part of

[3]We omit t_c when calculating the sampling rate of ADC and the duty cycle of other components, because t_c is 3 orders lower than t_T.

Table 1: Error of Serafin with a laptop or smartphone as signal source.

Task	Gesture		Activity		Respiration	
Source	Laptop	Smartphone	Laptop	Smartphone	Laptop	Smartphone
Error	5.8%	5.4%	4.7%	4.3%	4.2%	4.2%

Table 2: Power consumption (unit: μW) of each component. (B): Beacon as signal source. (L/S): Laptop or smartphone as signal source.

Sensing Task	Wake-up Circuit	ADC Samp	MCU 2MHz	MCU Sleep	Trans Circuit	MCU 80MHz	Overall Power
Gesture (B)	189	5	163	192	129	97	775
Activity (B)	189	5	163	192	129	62	740
Respiration (B)	189	5	163	192	129	0.34	678
Gesture (L/S)	189	1	41	228	34	97	590
Activity (L/S)	189	1	41	228	34	62	555
Respiration (L/S)	189	1	41	228	34	0.34	493
Localization (B)	189	10	326	144	258	11.8	939
Liquid RI (B)	189	5	163	192	129	0.96	679
Liquid Type (B)	189	5	163	192	129	13.4	691

the inter-packet duration, and t_r is a random variable that uniformly distributed over $\{n \times 1ms : 0 \leq n \leq 10, n \in \mathbb{Z}\}$.

- The transformation circuit is only powered when ADC sampling is required. Its duty cycle is thus $\mathbb{E}[t_l/(t_l+t_f+t_r)]$.
- The MCU is in its active mode with 2MHz system clock frequency when Bluetooth advertising signal appears. Besides, it also occasionally enters this mode when other signals trigger the wake-up circuit when Serafin is in (IC, SLP). Nonetheless, its duty cycle is smaller than $\mathbb{E}[(t_r + t_l)/(t_f + t_r + t_l)]$. We set such value as the duty cycle of this mode for power consumption analysis.
- The duty cycle of MCU sleep is 1-$\mathbb{E}[(t_r + t_l)/(t_f + t_r + t_l)]$.

Power consumption breakdown. Given the above analysis, we provide the power consumption breakdown of each component in executing different sensing tasks. In our experiments, the (t_T, t_c) pair of each task is: $(2\text{s}, 20.3ms)$ for gesture recognition, $(3\text{s}, 19.5ms)$ for activity recognition, $(20\text{s}, 0.7ms)$ for respiration sensing, $(0.3\text{s}, 3.7ms)$ for localization, $(1\text{s}, 0.1ms)$ for refractive index measurement, and $(1\text{s}, 1.4ms)$ for liquid classification. Note that for the localization task, the duty cycles of the MCU active at 2MHz system clock frequency, the transform circuit, and the ADC double because we use two beacons. And we have $t_f = 20ms$ with the beacon as the signal source, and $t_f = 100ms$ with the laptop or smartphone as the signal source. We calculate the average power consumption of each component for each task in Tab. 2. Serafin completes these tasks with the total power consumption ranging from only 493μW to 939μW.

8 Related Work

Sub-mW RF sensors. Thus far, a set of prior works have devoted themselves to design sub-mW RF sensors. Among them, a thread of works [29–35] propose to utilize commercial RFID tags or design dedicated backscatter tags to execute various RF sensing tasks, including localization [29], eye sensing [30, 31], gas sensing [32], sweat sensing [33], high-rate motion sensing [34], egg sensing [35], and so on. Although these tags are ultra-low-power themselves, they require nearby dedicated powerful readers [36, 37] for signal processing and result computation. Such working mode makes these tags unable to bring transparent RF sensing to mobile and IoT devices as Serafin does.

There exist another line of works that propose full-stack sub-mW RF sensors [5, 6]. As far as we know, the pioneer work AllSee [5] is the first one along this line. AllSee employs envelope detector to extract the amplitude of the received signal, and computes sensing results based on amplitude features. However, AllSee is specifically tailored to only one task of gesture recognition. Another work NeuroRadar [6] is a radar-based RF sensor that supports 2 tasks. Its power consumption is sub-mW-level for gesture recognition, but increases to around 2mW for localization. Besides, the neuromorphic processor of NeuroRadar is prohibitively expensive, preventing the wide-spread application of NeuroRadar. In fact, the primary reason that makes AllSee and NeuroRadar lack of versatility lies in their inability to extract the phase difference. Among all sub-mW RF sensors, Serafin is the only one that jointly achieves full stackness and versatility.

$\Omega(m\text{W})$ RF sensors. The power consumption of RF sensors in the majority of existing works are far above sub-mW level. The WiFi sensing works [2, 38–46] utilize WiFi NICs (e.g., Intel 5300 [47]), sniffers (e.g., Netgear A6210 [48]), chipsets (e.g., BCM43455 [49]), or customized platform [46] for sensing. The power consumption of these devices are at least 100mW-level. Recently, many works [50–58] propose to utilize wideband mmWave or UWB signals for RF sensing. They utilize a variety of RF sensors including mmWave radars [59], UWB chips [60, 61], mmWave communication devices [62], or customized platforms [7, 51]. Albeit powerful, the power consumption of these devices range from 100mW-level to W-level. Other works also employ Bluetooth [63], LTE [64], GPS [65], or LoRa [66, 67] for RF sensing. The power consumption of the RF radios [36, 37, 68, 69] in these works reach W-level. Serafin consumes 3-4 orders of magnitude lower power compared with existing $\Omega(m\text{W})$ RF sensors, but is versatile enough to support most sensing tasks that can be executed by them.

Low-power downlink communication. Another set of related works [70–73] aim to achieve low-power downlink communication by demodulating OFDM [71], CSS [70, 72], and DSSS [73] modulated signals. Similar to Serafin, the work Saiyan [70] also adopts the paradigm of RF computing by directly manipulating RF signals in analog domain to obtain the properties of signals. However, [70–73] can neither extract phase difference, nor support the computation (e.g., neural network inference) to obtain the sensing results as Serafin does. The wake-up circuits in [74–76] could be utilized to detect the arrival of downlink RF signal for downlink communication. However, compared with that in Serafin, these wake-up circuits either work in different frequency

band [74], or cannot simultaneously meet the sensitivity and power consumption requirements of Serafin [75, 76].

9 Discussion and Limitation

IC implementation. Although the current Serafin prototype consumes only sub-mW power and is already relatively compact, implementing Serafin with integrated circuit (IC) will further reduce its power consumption and form factor. Specifically, the power consumption of the transformation circuit and the wake-up circuit can reduce around 50 times and 3 times respectively if implemented in 65 nm CMOS technology [77, 78], making the total power consumption of completing the tasks in Tab. 2 ranging from 334μW to 599μW. The IC version of Serafin could also support tens of receiving antennas while still consuming only sub-mW power. Such a lager antenna array would clearly boost Serafin's performance on many sensing tasks. Besides, the size of Serafin in IC version will be only chip-level, which facilitates integrating Serafin on small mobile and IoT devices with negligible impact on their form factors.

Pre-measuring. For a given manufactured Serafin RF front end, the pre-measuring process described in § 3.2 only needs to be executed once. In mass production, for the Serafin RF front ends built with the same batch of capacitors and inductors, we could pre-measure only a few of them and obtain confident estimations of p_1, p_2, and ψ, which avoids the labor-intensive pre-measuring for every single one.

Eliminating out-of-band interference. Adding a bandpass filter centered at around 2.4GHz after Serafin's each antenna is promising to eliminate the out-of-band interference (e.g., continuous LTE downlink transmission). However, the insertion loss brought by the filters will slightly degrade the sensitivity of Serafin's RF front end.

Impact of multipath. One limitation of Serafin is that in multipath-rich environments, it cannot distinguish the phase difference of signals propagating in each path. Although such phase difference could be potentially obtained by utilizing the multipath spectrum estimation algorithm (e.g., MUSIC [79]), in practice, achieving this will increase Serafin's power consumption to mW-level. Specifically, such kind of multipath spectrum estimation algorithm requires to combine Serafin's extracted strengths and phase differences of the received superposition signals to form the input signal vector. However, the current number of receiving antennas in Serafin is only 3, and consequently the signal vector is of length 3 and is too short for multipath spectrum estimation algorithm. Obtaining a longer input signal vector requires to add transformation circuits and will increase the power consumption. It is also notable that unlike wideband RF sensors, Serafin uses near single-tone signal for sensing and cannot generate accurate power-delay profiles to characterize multipath channels. Thus, Serafin cannot conduct localization in complex indoor environments, which requires accurate phase difference of signals propagating in each path.

Despite the aforementioned limitation, utilizing the phase differences of the superposition signals extracted by Serafin is adequate to obtain reliable results for the tasks of gesture and activity recognition, and respiration sensing in complex indoor environments, as elaborated in Sec. 7.3.1-7.3.3.

Sensing signal type. Currently, Serafin uses Bluetooth advertising signal as its sensing signal. This restricts Serafin to be applied in environments with devices that periodically transmit Bluetooth advertising signals (e.g., beacon, laptop), or to be integrated into devices with built-in Bluetooth modules (e.g., smartphone). It would be beneficial to make Serafin compatible with other types of signals besides Bluetooth advertising signals for sensing. In fact, it is promising for Serafin to utilize other near single-tone signals (e.g., DBPSK modulated 802.11b signal) with no hardware modification.

Application scope. Despite being versatile, Serafin is unsuitable for conducting two kinds of sensing tasks. The first kind of tasks are those requiring the phase variation of the sensing signal received by a single antenna, such as detecting millimeter-level displacement. Serafin cannot extract such phase variation required by those tasks. The second kind of tasks are those requiring high-rate (>50Hz) phase difference samples, such as high-rate vibration sensing. Serafin cannot support those tasks because its frequency of obtaining phase difference is less than 50Hz, which is limited by the minimum transmission interval of Bluetooth advertising packets. We believe in the aforementioned tasks traditional RF sensors remain better alternatives than Serafin.

Sensing range. Limited by the sensitivity of the envelope detectors, the maximum sensing range between Serafin and the Bluetooth beacon whose transmission power is 20dBm is around 3.5m, as indicated by the experiment in Sec. 7.2.1. Our additional experiments indicate that the maximum sensing range drops to 1.8m when the laptop is the signal source, and drops to 0.3m when the smartphone is the signal source. Adding amplifiers will improve the sensing range, but will also increase the power consumption of Serafin.

10 Conclusion

This paper presents Serafin, the first full-stack, sub-mW and versatile Bluetooth-enabled RF sensor that brings transparent RF sensing to mobile and IoT device. Serafin stands out from existing sub-mW counterparts by simultaneously enjoying full stackness and versatility, and consumes 3-4 orders of magnitude lower power compared with existing counterparts that possess these two merits.

Acknowledgments

This work was supported by NSF China (No. U21A20519, 62372288, 62202298, W2412089).

References

[1] J. Lien, N. Gillian, M. E. Karagozler, P. Amihood, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev, "Soli: ubiquitous gesture sensing with millimeter wave radar," *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 142:1–142:19, 2016.

[2] J. Hu, T. Zheng, Z. Chen, H. Wang, and J. Luo, "Muse-fi: Contactless muti-person sensing exploiting near-field wi-fi channel variation," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2023.

[3] "Google pixel 4 xl battery issue: it's even draining when the phone is idle," https://www.valuewalk.com/google-pixel-4-battery-issue-idle/.

[4] "Soli not on Pixel 9," https://www.androidpolice.com/google-soli-appears-in-nest-learning-thermostat/.

[5] B. Kellogg, V. Talla, and S. Gollakota, "Bringing gesture recognition to all devices," in *Proceedings of the 11th USENIX Symposium on Networked Systems Design and Implementation, NSDI*, 2014.

[6] K. Zheng, K. Qian, T. Woodford, and X. Zhang, "Neuroradar: A neuromorphic radar sensor for low-power iot systems," in *Proceedings of the 21th ACM Conference on Embedded Networked Sensor Systems, Sensys*, 2023.

[7] Z. Chen, T. Zheng, and J. Luo, "Octopus: a practical and versatile wideband MIMO sensing platform," in *Proceedings of the 27th ACM Annual International Conference on Mobile Computing and Networking, MobiCom*, 2021.

[8] "PD2425J5050S2HF," https://cdn.ttm.com/repository/products/wireless-xinger/power-dividers/PD2425J5050S2HF/PD2425J5050S2HF.pdf.

[9] "TXLINE," https://www.cadence.com/en_US/home/tools/system-analysis/rf-microwave-design/awr-tx-line.html.

[10] "ADS," https://www.keysight.com/us/en/products/software/pathwave-design-software/pathwave-advanced-design-system.html.

[11] "LTC5530," https://www.analog.com/en/products/ltc5530.html.

[12] "HM8135," https://www.tequipment.net/Hameg/HM8135/.

[13] "SMS7630-005LF," https://www.skyworksinc.com/.

[14] "MAX9620," https://www.analog.com/en/products/max9620.html.

[15] "RS8521," http://www.grandchip.com/uploads/products/RS8522.pdf.

[16] "MAX9075," https://www.analog.com/en/products/max9075.html.

[17] "Tflite Converter," https://www.tensorflow.org/lite/models/convert.

[18] "STM32L467RG," https://www.st.com/en/microcontrollers-microprocessors/stm32l476rg.html.

[19] "CC1352P," https://www.ti.com/lit/ds/symlink/cc1352p.pdf.

[20] "Thinkpad X1 Carbon," https://www.lenovo.com/us/en/d/thinkpad-x1-carbon/.

[21] "Xiaomi 8 SE," https://www.mi.com/mi8i.

[22] "AD8302," https://www.analog.com/en/products/ad8302.html.

[23] X. Zhang, P. Cui, R. Xu, L. Zhou, Y. He, and Z. Shen, "Deep stable learning for out-of-distribution generalization," in *Proceedings of the 34th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2021.

[24] "JZ-130R," https://www.kiuzou.com/products-0-50-p.html.

[25] "Livox Horizon," https://www.livoxtech.com/3296f540ecf5458a8829e01cf429798e/assets/horizon.

[26] "Intel NUC 11," https://ark.intel.com/content/www/us/en/ark/products/121572/intel-nuc-11-pro-kit-nuc11tnhi30z.html.

[27] J. Lin and F. Zhang, "Loam livox: A fast, robust, high-precision lidar odometry and mapping package for lidars of small fov," in *Proceedings of the 37th IEEE International Conference on Robotics and Automation, ICRA*, 2020.

[28] R. Augustine, S. Raman, and A. Rydberg, "Relative permittivity measurements of et–oh and mt–oh mixtures for calibration standards in 1–15 ghz range," *Electronics letters*, vol. 50, no. 5, pp. 358–359, 2014.

[29] L. Dodds, I. Perper, A. Eid, and F. Adib, "A handheld fine-grained RFID localization system with complex-controlled polarization," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2023.

[30] L. Li, Y. Xie, J. Xiong, Z. Hou, Y. Zhang, Q. We, F. Wang, D. Fang, and X. Chen, "Smartlens: sensing eye activities using zero-power contact lens," in *Proceedings of the 28th ACM Annual International Conference on Mobile Computing and Networking, MobiCom*, 2022.

[31] L. Li, B. Shang, Y. Wu, J. Xiong, X. Chen, and Y. Xie, "Cyclops: A nanomaterial-based, battery-free intraocular pressure (IOP) monitoring system inside contact lens," in *Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation, NSDI*, 2024.

[32] X. Sun, J. Xiong, C. Feng, X. Li, J. Zhang, B. Li, D. Fang, and X. Chen, "Gastag: A gas sensing paradigm using graphene-based tags," in *Proceedings of the 30th ACM Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[33] W. Jiao, Y. Li, X. Xi, J. Wang, D. Fang, and X. Chen, "Bioscatter: Low-power sweat sensing with backscatter," in *Proceedings of the 21st ACM Annual International Conference on Mobile Systems, Applications and Services, MobiSys*, 2023.

[34] M. Xie, M. Jin, F. Zhu, Y. Zhang, X. Tian, X. Wang, and C. Zhou, "Enabling high-rate backscatter sensing at scale," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[35] Z. Sun, T. Ni, Y. Chen, D. Duan, K. Liu, and W. Xu, "Rf-egg: An rf solution for fine-grained multi-target and multi-task egg incubation sensing," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[36] "USRP B210," https://www.ettus.com/all-products/ub210-kit/.

[37] "USRP N210," https://www.ettus.com/all-products/un210-kit/.

[38] Y. Xie, J. Xiong, M. Li, and K. Jamieson, "md-track: Leveraging multi-dimensionality for passive indoor wi-fi tracking," in *Proceedings of the 25th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2019.

[39] S. M. Hernandez and E. Bulut, "Wifi sensing on the edge: Signal processing techniques and challenges for real-world systems," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 46–76, 2023.

[40] S. Ji, Y. Xie, and M. Li, "Sifall: Practical online fall detection with RF sensing," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems, SenSys*, 2022.

[41] Z. Yao, X. Wang, K. Niu, R. Zheng, J. Wang, and D. Zhang, "Wiprofile: Unlocking diffraction effects for sub-centimeter target profiling using commodity wifi devices," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[42] R. S. Ayyalasomayajula, A. Arun, C. Wu, S. Sharma, A. R. Sethi, D. Vasisht, and D. Bharadia, "Deep learning based wireless localization for indoor navigation," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2020.

[43] D. Zhang, J. Wang, J. Jang, J. Zhang, and S. Kumar, "On the feasibility of wi-fi based material sensing," in *Proceedings of the 25th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2019.

[44] Y. He, J. Liu, M. Li, G. Yu, J. Han, and K. Ren, "Sencom: Integrated sensing and communication with practical wifi," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2023.

[45] C. Wu, X. Huang, J. Huang, and G. Xing, "Enabling ubiquitous wifi sensing with beamforming reports," in *Proceedings of the ACM SIGCOMM 2023 Conference, SIGCOMM*, 2023.

[46] K. Song, Q. Wang, S. Zhang, and H. Zeng, "Siwis: Fine-grained human detection using single wifi device," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[47] "Intel WiFi Link 5300," https://www.intel.com/content/www/us/en/support.html.

[48] "A6210," https://www.netgear.com/support/product/a6210/.

[49] BCM43455, "Single-chip 5g wifi ieee 802.11ac mac/baseband/ radio with integrated bluetooth 4.1 and fm receiver," https://www.digikey.com/htmldatasheets/production/1955355/0/0/1/BCM43455.pdf.

[50] X. Zhang, D. Zhang, Y. Xie, D. Wu, Y. Li, and D. Zhang, "Waffle: A waterproof mmwave-based human sensing system inside bathrooms with running water," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 7, no. 4, pp. 1–29, 2023.

[51] R. Zhao, T. Woodford, T. Wei, K. Qian, and X. Zhang, "M-cube: a millimeter-wave massive MIMO software radio," in *Proceedings of the 26th ACM Annual International Conference on Mobile Computing and Networking, MobiCom*, 2020.

[52] H. Lai, G. Luo, Y. Liu, and M. Zhao, "Enabling visual recognition at radio frequency," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[53] F. Zhang, J. Xiong, Z. Chang, J. Ma, and D. Zhang, "Mobi^{2}sense: empowering wireless sensing with mobility," in *Proceedings of the 28th ACM Annual International Conference on Mobile Computing and Networking, MobiCom*, 2022.

[54] N. Mehrotra, D. Pandey, A. Prabhakara, Y. Liu, S. Kumar, and A. Sabarwal, "Hydra: Exploiting multi-bounce scattering for beyond-field-of-view mmwave radar," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[55] H. Zhou, K. Yuan, M. Gowda, L. Qiu, and J. Xiong, "Rethinking orientation estimation with smartphone-equipped ultra-wideband chips," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[56] E. Sie, Z. Liu, and D. Vasisht, "Batmobility: Towards flying without seeing for autonomous drones," in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2023.

[57] U. Ha, S. Assana, and F. Adib, "Contactless seismocardiography via deep learning radars," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2020.

[58] E. Soltanaghaei, A. Prabhakara, A. Balanuta, M. G. Anderson, J. M. Rabaey, S. Kumar, and A. G. Rowe, "Millimetro: mmwave retro-reflective tags for accurate, long range localization," in *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking,MobiCom*, 2021.

[59] "IWR1642," https://www.ti.com/product/IWR1642/.

[60] "NOVELDA X7," https://novelda.com/x7-radar-direct.

[61] "TREK1000," https://datasheet.octopart.com/TREK1000-Decawave-datasheet-98959800.pdf.

[62] "QCA6428," https://www.qualcomm.com/products/internet-of-things/networking/wi-fi-networks/qca6428.

[63] R. S. Ayyalasomayajula, D. Vasisht, and D. Bharadia, "Bloc: Csi-based accurate localization for BLE tags," in *Proceedings of the 14th ACM International Conference on emerging Networking EXperiments and Technologies, CoNEXT*, 2018.

[64] Y. Feng, Y. Xie, D. Ganesan, and J. Xiong, "Lte-based low-cost and low-power soil moisture sensing," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems, SenSys*, 2022.

[65] H. Dong, M. Cui, N. Wang, L. Qiu, J. Xiong, and W. Wang, "Gpsense: Passive sensing with pervasive gps signals," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking, MobiCom*, 2024.

[66] B. Xie, D. Ganesan, and J. Xiong, "Embracing lora sensing with device mobility," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems, SenSys*, 2022.

[67] B. Xie, M. Cui, D. Ganesan, X. Chen, and J. Xiong, "Boosting the long range sensing potential of lora," in *Proceedings of the 21st ACM Annual International Conference on Mobile Systems, Applications and Services, MobiSys*, 2023.

[68] "RTL-SDR," https://www.rtl-sdr.com/about-rtl-sdr/.

[69] "u-blox F9 high precision GNSS module," https://content.u-blox.com.

[70] X. Guo, L. Shangguan, Y. He, N. Jing, J. Zhang, H. Jiang, and Y. Liu, "Saiyan: Design and implementation of a low-power demodulator for lora backscatter systems," in *Proceedings of the 19th USENIX Symposium on Networked Systems Design and Implementation, NSDI*, 2022.

[71] F. Zhu, L. Feng, M. Jin, X. Tian, X. Wang, and C. Zhou, "Towards ultra-low power OFDMA downlink demodulation," in *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems, SenSys*, 2022.

[72] Y. Song, L. Lu, J. Wang, C. Zhang, H. Zheng, S. Yang, J. Han, and J. Li, "μmote: Enabling passive chirp de-spreading and μw-level long-range downlink for backscatter devices," in *Proceedings of the 20th USENIX Symposium on Networked Systems Design and Implementation, NSDI*, 2023.

[73] S. Li, H. Zheng, C. Zhang, Y. Song, S. Yang, M. Chen, L. Lu, and M. Li, "Passive DSSS: empowering the downlink communication for backscatter systems," in *Proceedings of the 19th USENIX Symposium on Networked Systems Design and Implementation, NSDI*, 2022.

[74] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. M. Popovici, and L. Benini, "Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver," *IEEE Trans. Ind. Informatics*, vol. 12, no. 2, pp. 633–644, 2016.

[75] P. Zhang, D. Bharadia, K. R. Joshi, and S. Katti, "Hitchhike: Practical backscatter using commodity wifi," in *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems, SenSys*, 2016.

[76] X. Na, X. Guo, Z. Yu, J. Zhang, Y. He, and Y. Liu, "Leggiero: Analog wifi backscatter with payload transparency," in *Proceedings of the 21st Annual International Conference on Mobile Systems, MobiSys*, 2023, pp. 436–449.

[77] S. B. Idres, M. El-Nozahi, and H. F. Ragai, "A noise cancelling envelope detector for low power wireless sensor applications," in *Proceedings of the 47th IEEE International Symposium on Circuits and Systems, ISCAS*, 2015.

[78] D. Baez-Villegas and J. Silva-Martinez, "Quasi rail-to-rail very low-voltage opamp with a single pmos input differential pair," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 11, pp. 1175–1179, 2006.

[79] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.